IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 2 April 2023, accepted 9 June 2023, date of publication 19 June 2023, date of current version 13 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3287491

RESEARCH ARTICLE

CARINA Project: Visual Perception Systems Applied for Autonomous Vehicles and Advanced Driver Assistance Systems (ADAS)

DIEGO RENAN BRUNO[1], RAFAEL A. BERRI[2], FELIPE M. BARBOSA[1], AND FERNANDO S. OSÓRIO[1], (Member, IEEE)

[1]Institute of Mathematical and Computer Sciences (ICMC), University of São Paulo (USP), São Paulo 05508-220, Brazil
[2]Federal University of Rio Grande (FURG), Rio Grande 96203-900, Brazil

Corresponding author: Diego Renan Bruno (diego.renan.bruno@gmail.com)

The work of Diego Renan Bruno and Felipe M. Barbosa was supported by the University of São Paulo at Sao Carlos through the Graduate Program funded by FUNDEP and ROTA2030.

ABSTRACT Autonomous mobile robots use computational techniques of great complexity so that to allow navigation in various types of dynamic environments, avoiding collisions with obstacles and always seeking to optimize the best route, ultimately enabling them to operate in a safe and precise manner. In order for navigation at this level to be possible, a variety of computer vision and intelligent sensing techniques are used. The potential of an intelligent computer vision system to detect and predict the actions of dynamic agents on the streets is applied to increase traffic safety with intelligent robotic vehicles. In this paper we present a systematic review of computer vision models for the detection and tracking of obstacles in traffic environments. Specifically, we cover works involving 2D and 3D (stereo vision) data fusion for both internal and external perception, as well as current trends regarding efficient model design and temporally-aware architectures. We provide a thorough discussion on the main positive and negative points of the state-of-the-art in Visual Robotic Attention, as well as share our experience and contributions in applying visual perception for external obstacle detection and tracking, and internal (driver) monitoring. The results presented should serve as a compilation of the history of visual perception for autonomous mobile robots (specifically, Advanced Driver Assistance Systems (ADAS) and Autonomous Vehicles), thus providing the reader with a comprehensive basis on both the main contributions and the state-of-the-art in the field.

INDEX TERMS Autonomous vehicles, computer vision, deep learning, obstacle detection and classification.

I. INTRODUCTION

Autonomous mobile robots use computational techniques of great complexity so that it is possible to safely and accurately navigate in various types of dynamic environments, avoiding collisions with obstacles and always seeking to optimize the best route. An area of great interest in mobile robotics is closely linked to navigation of robotic vehicles. These vehicles, embedded with a complex stack of modules for perception, planning and actuation, can assist the driver in various traffic conditions and, ideally, navigate the environment without the need for human intervention, and within the traffic laws. Therefore, Intelligent Robotic Vehicles are mainly applied for the benefit of the reduction of traffic accidents, by compensating for human faults and recklessness.

In order for both diver assistive technologies and navigation at autonomous level to be possible, a variety of Computer Vision, Artificial Intelligence, Automation methods are applied. When considering sensing techniques, visual perception is of utmost importance, since it can provide rich $2D$ and $3D$ environmental information, at considerably reduced costs.

The main objective of this research was the study and presentation of a set of Computer Vision and Artificial Intelligence techniques and methods, so that to cover important contributions to $2D$ and $3D$-based visual perception – as well

The associate editor coordinating the review of this manuscript and approving it for publication was Wei Wei.

as their fusion. Besides that, we share our experience in the field by presenting works developed by our research group at the Mobile Robotics Lab (LRM).

A. ADVANCED DRIVER ASSISTANCE SYSTEMS (ADAS)

According to the SAE Levels of Driving Automation™ (Fig. 1), vehicle automation technologies are classified into increasing levels of driver assistance and, consequently, decreasing levels of driver intervention. Advanced Driver Assistance Systems (ADAS) are technologies intended to assist the driver in the driving task, corresponding from levels 0 to 4 in the SAE scale. Several of these systems are based on Computer Vision [1]. The amplitude of the ADAS goes beyond the vehicle itself, as it must be defined as a set of systems and subsystems that allow automation of highways [2]. Thus, ADAS can improve vehicle safety and, consequently, road safety, reducing or even eliminating possible driver errors [3]. Since the beginning of ADAS adoption at vehicles is observed a significant reduction of the fatal victims in road accidents. In European countries, for example, there was a reduction between 30% and 40% in fatal accidents as a result of the implementation of such systems [4].

ADAS can be divided into subcategories according to their role in supporting management. According to [6], the categorization is the following: Lateral control, Longitudinal control, Parking/reversing aids, Vision enhancement, Driver monitoring, Pre-crash systems, and Road surface/low-friction warning.

Lateral Control systems prevent unintentional lane departures or dangerous lane changes. Examples of these types of systems are shown below:

- Lane Departure Warning helps the driver to keep the vehicle within the lane, warning when the vehicle is leaving the current lane, in the way, inappropriately or dangerously. This system checks for nearby objects and the activation of signal lights (lane change intention) [7]. Therefore, it is designed to minimize collisions mainly caused by driver error, distractions or drowsiness.
- The Blind Spot system monitors the area to the side of the vehicle, mainly the region where drivers cannot see through the rearview mirrors [8]. If the system detects a vehicle in this region, acoustic and/or visual signals alert the driver of the risk of collision.

Longitudinal Control encompasses speed control, distance control, and reaction times. Examples of these types of applications are:

- Adaptive Cruise Control automatically adjusts the distance to the vehicle ahead and the speed, using sensors to measure the longitudinal distance between vehicles [9]. Its purpose is to improve driving comfort, reduce traffic accidents and increase traffic flow (average speed).
- High-beam Assist detects the light of vehicles traveling in the opposite direction or those in front, alternating between high and medium beam [10].
- Traffic Sign Recognition aims at detecting and subsequently classifying traffic signs that define dangers and limitations on roads [11]. With this system, it is possible to recognize, for example, the speed limit or sharp curves ahead.

Parking/reversing aids are obstacle detection systems in low-speed situations. Park Assist is an example of an approach in this category, which allows the vehicle to park itself, involving automatic steering and speed control [12].

Vision Enhancement Systems support the driver in situations of reduced visibility. An example is Night Vision, whose objective is to increase the driver's ability to see obstacles during the dark hours of the day, uses cameras sensitive to infrared radiation [9]. While most car headlights can illuminate the road about 60 meters ahead of the vehicle, this system allows drivers to obtain traffic information from up to 150 meters away.

Driver monitoring focuses on the physiological and behavioral state of the driver. With this kind of system, the vehicle can analyze the driving by a human driver and determines if it is a security performance. Driving state monitoring can be divided into two main branches the detection of distractions and the identification of drowsiness [13, p. 23].

In Road surface/low-friction warning, the main objective is to alert the driver in case of poor road conditions. The system can issue warnings or be directly related to the speed control system, for example, helping the driver to maintain the appropriate speed for the current road conditions, or even detecting potholes in the road [14].

Pre-crash systems aim to avoid or minimize the possible problems of an ongoing accident. They act when the driver may no longer be able to react. Examples of this category of ADAS include:

- Airbags are bags inflated very quickly with air when the vehicle [15] collides. The most common airbags are the front and side airbags that protect the occupants of the vehicle.
- The Forward Collision Warning is the system that provides warnings (visual, auditory or vibration) to the driver when a probable imminent accident with the vehicle in front is detected [16].
- Object Detection aims to detect objects in the path of the vehicle. An example of this type of approach is pedestrian detection in order to alert the conduit. In some case it can even perform automatic braking or even deploy external airbags (in the case of an unavoidable collision) [17].

Figure 2 shows the possible locations of sensors or cameras for video acquisition of the ADAS. It can also be seen that the Lane Departure Warning, High Beam Assist, Traffic Sign Recognition, Forward Collision Warning and Object Detection systems can use the same video input, starting from a front camera. However, each of the systems has different resolution and processing requirements.

Even with the massive adoption of ADAS in vehicles, it is among the top 8 causes of death for people on the

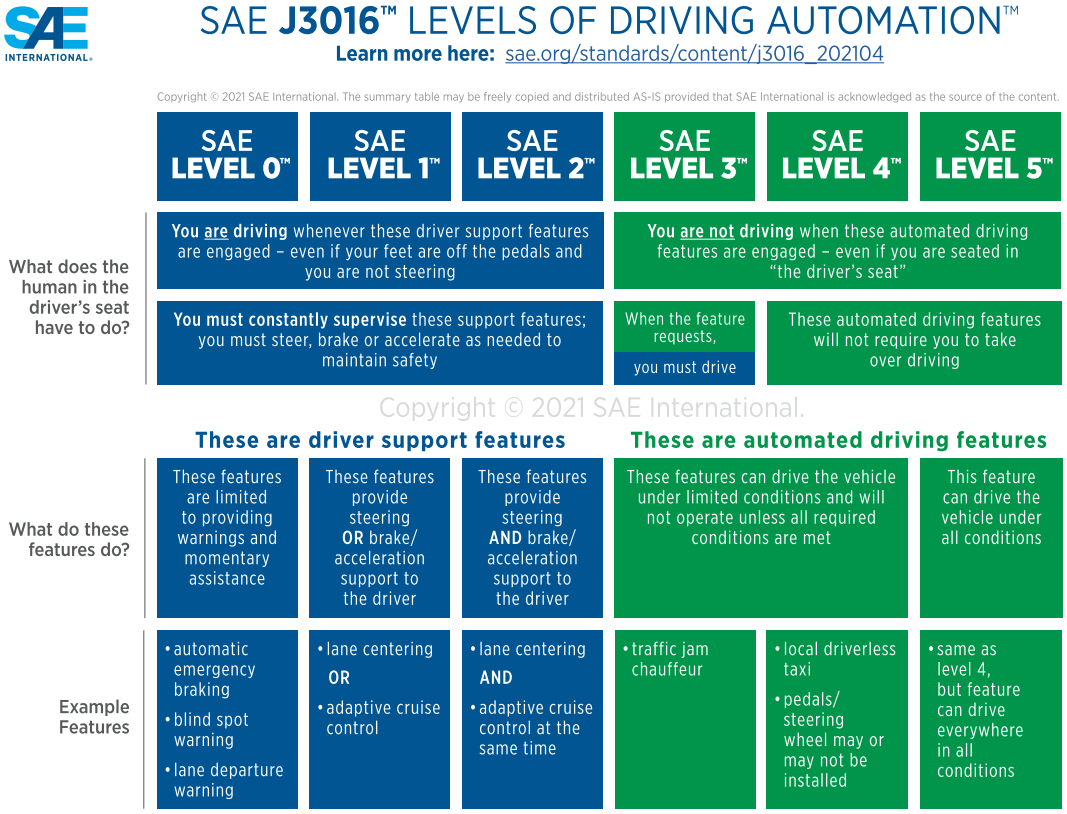


FIGURE 1. SAE levels of driving automation™ [5].

planet [19]. Looking at fatal traffic accidents alone, 95% of these are caused by human error [20]. The three main causes of human errors in traffic are drunkenness, drowsiness and driver distraction in general [21].

The adoption of vehicles capable of moving autonomously is also a current research path. This adoption would reduce human error to zero, but creates machine errors, which certainly can be mitigated in order to increase road safety. A discussion on Autonomous vehicles takes place in the next section.

B. AUTONOMOUS VEHICLES

Autonomous vehicles correspond to SAE Level 5™ technologies in the SAE Levels of Driving Automation™ (Fig. 1), characterized by the vehicle being able to perform the driving task under all conditions, ideally not requiring the driver to take over.

These systems can contribute in diverse ways to the overall safety and mobility in urban and roadway environments. In first place, mobility can be increased for people who do not have the ability to drive a vehicle due to some type of restriction and wish to have accessibility for fast and safe locomotion in an urban environment. It also offers advances in convenience; for example, an autopilot system can support a driver who needs to rest on a long-distance trip - which also represents an increase in safety, by avoiding that the driver operates the vehicle under the influence of drowsiness.

As for safety, societal and economic aspects, Autonomous vehicles reduce the chances of traffic accidents caused by human errors. This, in turn, represents a reduction of traffic victims who, besides possibly becoming unable to work, may suffer from long-term injuries, with a considerable reduction in their quality of life. Finally, efficiency can be increased through the cooperation among autonomous vehicles in order to reduce traffic congestion.

There are several autonomous driving projects currently under development, such as the CaRINA [22] developed together with the ICMC/USP LRM and the *Google Self-Driving Car* [23], which show the great effort scientific development that has taken place in recent years towards more autonomous vehicles. However, as long as steering wheels remain in vehicles, even autonomous ones, there will still be the option for a human driver to request manual control of the vehicle. And this existence of the steering wheel is due to the fact that autonomous vehicles need an infrastructure prepared to operate autonomously and safely [24], so in certain places, it may not yet be possible to guarantee fully autonomous driving, requiring intervention and the driving of the vehicle by a human driver.

So, depending on the user's destination, the vehicle may not be able to operate autonomously all or part of the journey. Therefore, there is still much room for improvement in vehicular technologies, in order to adapt to scenarios where conditions are not yet viable for autonomous operation.

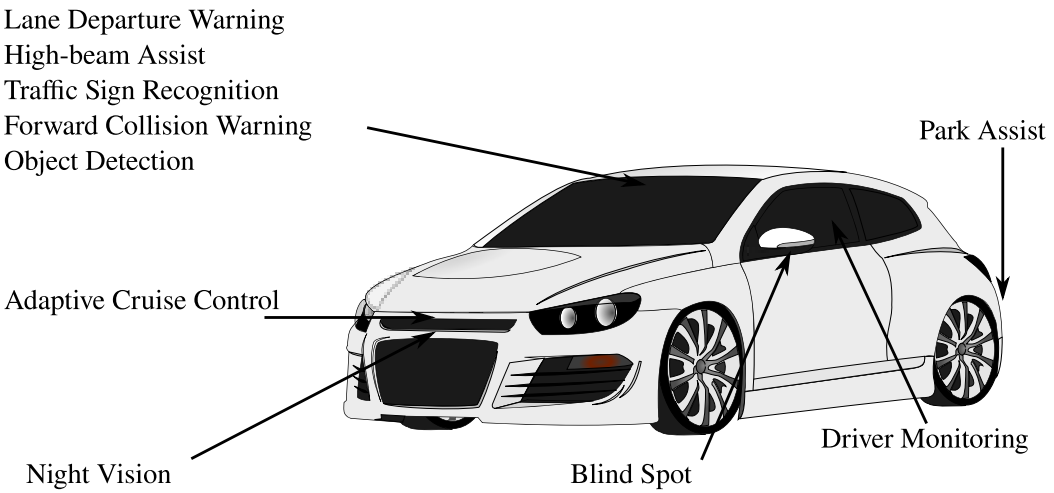


FIGURE 2. View of some ADAS sensors positions. Adapted from [18].

II. CaRINA PROJECT

CarINA is a robotic platform used in the development of perception, control and decision-making systems for autonomous and assistive navigation in urban environments. It has computational control of steering, acceleration and braking, and several sensors such as: GPS, IMU, cameras and lasers.

Collaboration project between LRM and the company Scania presents an autonomous truck fully developed in Brazil. The platform used within the university campus has a computational system that processes information obtained by sensors in real-time, allowing safe and efficient navigation.

When it comes to perception models, the CARINA project is focused on the development of algorithms for the detection of obstacles, traffic signs and waterways in urban environments from point clouds. Point clouds are created by various types of sensors such as stereo cameras, laser sensors (LIDAR 3D).

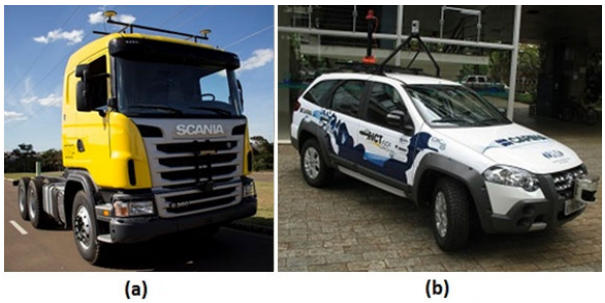


FIGURE 3. Autonomous vehicles of the LRM Lab. / USP. (a) First autonomous truck in Latin America; (a) The CaRINA 2 project (Intelligent robotic car for autonomous navigation project).

III. DEVELOPED WORKS

A. DETECTION AND CLASSIFICATION OF OBSTACLES USING 2D DATA

Images in 2*D* format images should contribute with great potential for a computer vision model applied to the obstacle detection task. They provide rich information on shapes, colors and textures and, also, through the union of two images, it is possible to estimate depth, very important for the detection of an obstacle. Currently, Deep Learning network models behave very well on this type of image, allowing, in addition to detection, the classification, segmentation and instantiation of an object.

In recent research, we have applied Deep Learning networks to detect and classify traffic signs in 2D images [25], [26], [27], [28], [29], [30], [31], taking advantage of the potential of this type of image for detection over long distances (approximately 100 meters), since in 3*D* images detection range (with good resolution) is limited to a few meters – approximately 30m.

Deep Learning networks make great contributions to the detection of objects in 2*D* images. There are two main structures for detecting objects in images: (1) three-stage detectors (detection, classification and instantiation): RCNN [32], Fast R-CNN [33] and Faster R-CNN [34]. (2) two-stage detectors (detection and classification): YOLO [35], SSD [36] and YOLO9000 [37].

Some networks have the potential to detect, segment and instantiate objects with great precision, however, generating false positives and false negatives for autonomous vehicle navigation. The DeepLab network (Figure 4) has a very important potential for the 2*D* computer vision area, also being able to support a system with 2*D* and 3*D* sensor fusion.

This type of approach involving data from monocular cameras (Figure 4) generally has a low computational and time cost, presenting good results in its general form. However, our researches [25], [26], [27], [28], [29], [30], [31] and that of other researchers in the field of intelligent vehicles, highlight the importance of 3*D* data for a better assessment of the navigable area. In the next section we present the works that stand out in applications involving the notion of depth of the scene.

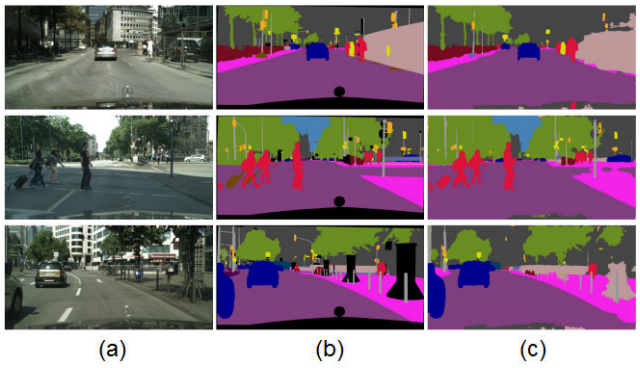


FIGURE 4. DeepLab: (*a*) Real Image, (*b*) Results 1 e (*c*) Conditional Random Field (*CRF*).

FIGURE 5. Detection of false traffic signs.

1) FALSE DETECTION

2D data is fundamental for a computer vision system to work effectively, however, there are big problems for vision models that use only this type of information, since the lack of depth information generates problems with object recognition and, also, problems with the identification of the position and tracking of the detected object. In the example of Figure 5, a problem of detection and recognition of traffic signs that come from real images can be observed, however, which do not represent real rules.

2) DETECTION AND LOCATION

The lack of notion of depth in 2D images also makes it impossible for a detected object to be evaluated in its location in relation to the vehicle. In the example of Figure 6, a situation can be observed where the vehicle detected two traffic signs, one for the vehicle that will turn right (30km-h), and another for the vehicle that continues on the straight road (80km-h). However, to declare which information must be obeyed, it is necessary to evaluate the position of the traffic signs in relation to the vehicle and its route, needing the coordinates (X, Y, Z).

Aiming at problems of this type (Figure 5 and Figure 6), our researchs were directed toward the fusion of 2D and 3D data for detection, classification and tracking of obstacles and traffic signs. A Computational vision model based on the YOLO [35] (Figure 5 and Figure 6) model do not support problems of this type when applied individually, so it is necessary to merge 2D and 3D data for greater robustness

FIGURE 6. Traffic signs detected in conflict.

of analysis of the perception system. In the next sections, the computer vision models developed will be presented in detail.

B. DETECTION AND CLASSIFICATION OF OBSTACLES USING 3D DATA

The images based on 3*D* data must contribute to a detection model, generating depth and geometry information for each detected object, these characteristics being very important to evaluate the tracking and shape of the obstacles. The fusion of 2*D* and 3*D* images allows a computer vision system to work with the potential attributes present in these different image formats, ensuring a more robust and accurate system.

In our research we have presented several works applied for the detection of vertical traffic signs in 3*D* images [25], [26], [27], [28], [29], [30], [31], enabling greater robustness for computer vision systems in favor of perception for intelligent robotic vehicles. Through the analysis of 3*D* images, it was possible to detect each traffic sign and evaluate its position in relation to the vehicle and other objects in the scene, also making it possible to perform a semantic analysis of the navigation environment to classify the relevance of each detected traffic sign.

In the work of Srivastava et al. [38], a system was developed capable of being trained with 3*D* images generated through 2*D* images. The model uses algorithms to generate images in depth, making it possible to work in the training phase with images with less noise compared to 3*D* images generated by physical sensors. In the testing phase, real 3*D* images available in the KITTI dataset were applied.

In a work by McCrae et al. [39], an obstacle detection system was developed in 3*D* images based on LIDARs sensing data. The system uses a recurring network that makes it possible to operate on a smaller volume of captured data and at a higher speed. However, the system does not have color and texture data of the objects, since it does not use cameras in its data collection.

In a work developed by Baek [40], an approach was proposed to detect and track curbs and obstacles by merging data from various sensors: sparse LiDAR data, a mono camera and ultrasonic sensors. The detection model is based on LiDAR 3D and a monocular camera sensor used to detect characteristics of candidate obstacles and remove false positives resulting from static and dynamic obstacles.

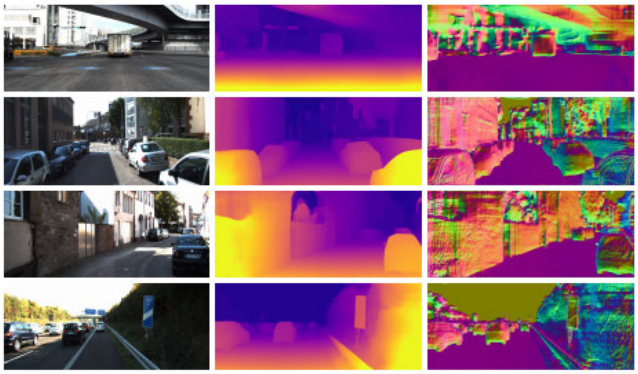


FIGURE 7. Depth estimation results on KITTI [42].

In the work of Weon et al. [41] an algorithm was developed to merge 3*D* LIDAR (Light Detection and Ranging) systems that receive objects detected in image sensors based on deep learning and object data in the form of 3*D* point clouds. However, depth data is used for depth and tracking estimates for each obstacle, while 2*D* data for each camera is applied to classify the obstacle class.

In summary, we have that the works that approach 3*D* image data, focus on the treatment of obstacle tracking through the notion of depth and also use this data to eliminate false positives. In some works, the fusion of sensors is applied, also making it possible to unite the potential of 2*D* and 3*D* images.

In the work of Guizilini et al. [42] it was proposed to use a self-supervised monocular depth estimate. In this work, a Geometric Unsupervised Domain Adaptation method (GUDA) makes it possible to learn an invariant domain representation using a multitasking objective, combining synthetic semantic supervision. The work also presented a good adaptation to the quality and quantity of synthetic data while improving depth prediction.

Through Figure 7, a depth analysis of the scene can be observed, preserving a scale learned from the direction model, improving the depth evolution if compared to standard fine-tuning methods [42].

One of the major problems for the area of 3D computer vision is related to the imperfection of data from these sensors, generating a reconstruction of the 3D object with flaws in its structure. In most cases, pixels are detected with errors in their depth, and this type of problem is very common for images generated by a stereo camera.

Through Figure 8, traffic signals that were detected in 2D images and reconstructed in their equivalent 3D image can be observed, allowing a more robust evaluation of the object. However, the good result was possible thanks to our $3D - CSD$ method and, also, because it is an image captured with a maximum distance of 30 meters in relation to the vehicle, more than this distance, the object starts to have problems in its reconstruction 3D.

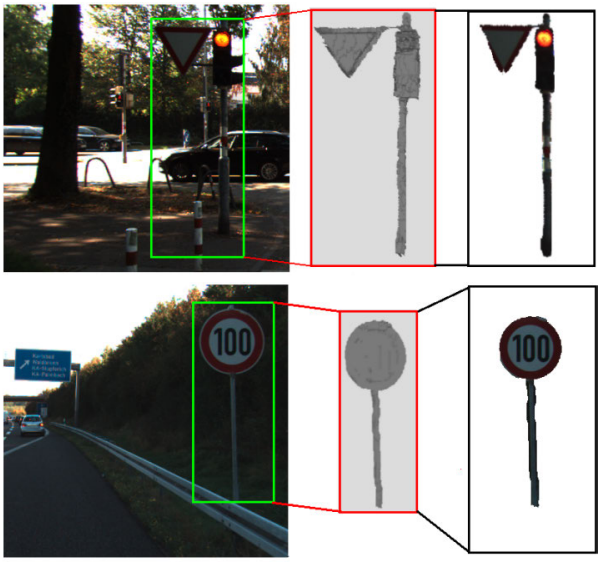


FIGURE 8. Depth estimation results on KITTI [42].

The fusion of 2D and 3D data enables a more robust analysis of objects in traffic, especially in relation to their shape and position in relation to the road and the vehicle.

As shown, there are several problems related to 2D computer vision for autonomous vehicles, and to solve these problems, computer vision with 3D data is essential for the proper functioning of the system.

In our research with the Mobile Robotics Laboratory (LRM) of the University of Sao Paulo, we have great experience in extracting and merging 2D and 3D data from cameras and LIDARs, aiming at a robust perception for the navigation of autonomous vehicles. In the next sections we present our main contributions.

C. EXTRACTION OF 3D FEATURES AND CLASSIFICATION OF 3D OBJECTS

In order to recognize the traffic signs, it is necessary to first detect and segment 3D objects in the environment scene. Next, an ANN is used to recognize the point cloud structure representing a 3D signature of these segmented objects, indicating whether it represents a traffic sign or not.

To detect and recognize the traffic signal object, you must first extract some characteristics (features). This should allow the ANN classifier to have enough 3D data to declare whether a traffic sign object / structure was detected or not.

To generate a standard entry with recognizable patterns for the ANN, the so-called ''object signature'', 3D-Contour Sample Distances method [43] has been applied. The features extracted by the 3D-CSD are based on the principle that each object class has a unique 3D outline appearance. For this to be possible, the point cloud of the three-dimensional object is converted into a vector representing the surface contour distances related to a central point of the object, providing a 3D signature recognizable by Machine Learning techniques.

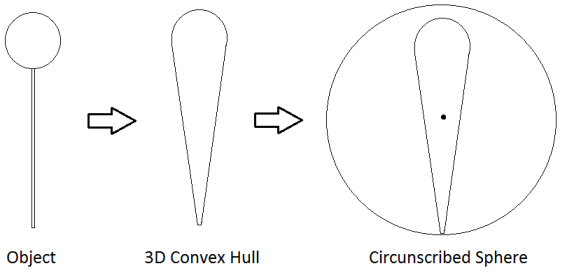


FIGURE 9. 2D view of sphere generation process.

The 3D-CSD descriptor is applied by measuring the distances from the center of mass of the object to the specific surface points of the object (''external contour'' or ''surface hull''), according to the selected and predefined key points in a circumscribed sphere. More simply, te 3D object is placed within a virtual sphere that has a predefined number of scattered (usually equally spaced) points on its surface. By means of virtual rays, the measurements (center of mass of the object) are drawn from the center of the sphere towards each of these key points scattered over the surface of the sphere. A video[1] was produced to explain this procedure. Thereafter, the measured distance values are interpolated to an appropriate (normalized) range and provided as input data from the machine learning classifier.

1) 3D-CSD FEATURE EXTRACTION

The first step is to estimate a surface mesh for the point cloud of the object by converting the sparse point cloud into a set of polygons (surface mesh) and obtaining the solid representation of the 3D shape object. We use the 3D Convex Hull algorithm for this task, so that the approximate contour of the object (shell of the object) can be obtained.

Next step, consider a sphere circumscribing the 3D Convex Hull (Figure 9). The sphere's center is positioned at the object's center of mass, and the radius is the furthest object's point from center. Figure 10 illustrates this process.

The next step is to select the various key points on the sphere. Each key point will correspond to a single distance measure and therefore a position in the vector of final distances. The key points should be equally distributed or, as a priority, the most representative areas of objects. For traffic signals, a generic strategy for distributing points along the surface of the sphere can be performed by defining an azimuth angle and constant altitude for the point distribution (Figure 4).

After selecting the points, the 3D-CSD (object distance vector) descriptor can be generated. For each key point p_i on the surface of the sphere, it is then considered a straight line from the center of p_i for the sphere. If this straight line crosses the surface of the object at a point d_i, calculate the Euclidean distance from point d_i to the center of mass of the object. If not, return -1.

[1]3D-CSD Feature Extraction explained in vídeo – Sphere, Virtual Rays and the Vector of Distances concepts – Available at: https://goo.gl/K5 × 5yB

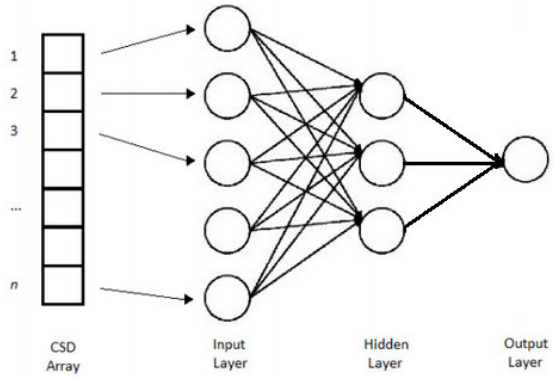


FIGURE 10. ANN architecture.

2) PATTERN RECOGNITION

In order to be able to recognize the 3D signature of the segmented object given by the 3D-CSD feature, an ANN Multi-Layer Perceptron (MLP) is used because of its good capabilities for capturing complex, non-linear underlying features with a high degree of accuracy. Figure 10 shows the ANN architecture.

The supervised learning method of ANN requires a set of training data that must be generated a priori. It is necessary to assemble a representative set of examples, composed of a large set of objects from different scenes, applying the descriptor 3D-CSD and labeling each example of object. In order to be able to recognize objects from different points of view (different orientations), additional examples of objects in different rotations are included in the training dataset.

D. ALGORITHM FOR DETECTION OF TRAFFIC SIGNS WITH 3D DATA

An Artificial Neural Network (ANN) with binary output has been trained with these various cases where boards (sign plates) and other elements can be found. The ANN was applied to solve this problem of classification and sign plate detection. For this to be possible, each type of case was modeled (Figure 11) based on the Velodyne LIDAR (Light Detection and Ranging) data and considering also a pair of stereo cameras, thus enabling the ANN network to respond if it is a board or an object that is not a sign plate.

In case of the neural network algorithm informs the system that a board (traffic sign candidate) has been detected in the environment, then a second classifier based on Deep Learning CNN is activated to classify the type of traffic sign that was detected in image RGB-D (Red, green, blue + Depth): maximum speed, cones for route deviation, stop, preferential, pedestrian or also other types of traffic signs.

In this situation, the 3D computer robotic vision system in conjunction with the ANN informs the detection and the

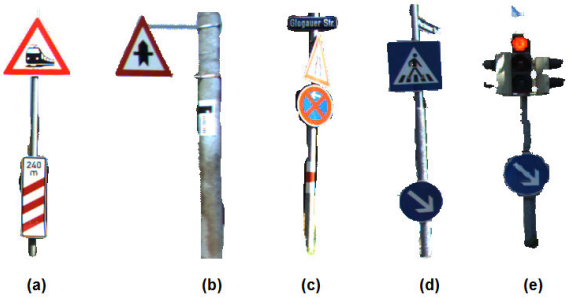


FIGURE 11. Traffic signs detected in different contexts (a) detected on a thin metallic pole (b) detected on a concrete pole (c) and (d) two sign plates detected on the same pole (e) traffic sign detected with a light signal.

coordinates where the possible traffic sign occurs (Figure 12). Given this information, the 2D image of the traffic sign detected is labeled and segmented, and submitted to the classifier based on the Deep Learning CNN, thus making it possible to identify which type of traffic sign was detected (Figure 12).

Deep Learnning CNN classification is applied to the 2D data (image-RGB), which takes advantage of the available textures and color information to recognize the different specific traffic signs. The recognition power of Deep CNN Networks (DCNN-inception V3), is also a big advantage, once they can recognize and classify images very well (including images with occlusion, damaged and in precarious lighting conditions), and on the other hand, this task can be very difficult if considering only the 3D data (shapes-Depth).

In Figure 13, the 3D perception (visual attention) and computer vision system can be observed in action generating the point cloud of one scene using the stereo camera data from the KITTI Dataset. Still, it can be observed that the traffic sign and other objects were detected with texture in the point cloud, these are some of the cases (Figure 13) trained in the ANN and used for detecting traffic signs objects.

Given Figure 13 it is possible to observe 2 types of segmented objects that are used to train our traffic signal detection system. Since one of them is not a traffic sign (b) and the other one is a sign used for stretches of highway (a).

Each segmented object (Figure 13) will be sent to a 3D feature extraction method. After that, the data extracted from each object will be applied as the input of an ANN-MLP for the classification of the detected object type.

By detecting a cone, our 3D vision system for auxiliary routes is activated, thus enabling the detection of an unmapped auxiliary route.

E. VISUAL ATTENTION

1) FUZZY VISUAL ATTENTION: DECISION MAKE

The decision-making process should be able to make an assessment of a finite information set, thus deciding the most appropriate action according to a set of fuzzy values. Making

FIGURE 12. Detection of cones in a scene using 3D data - real scene with data provided by stereo vision camera system.

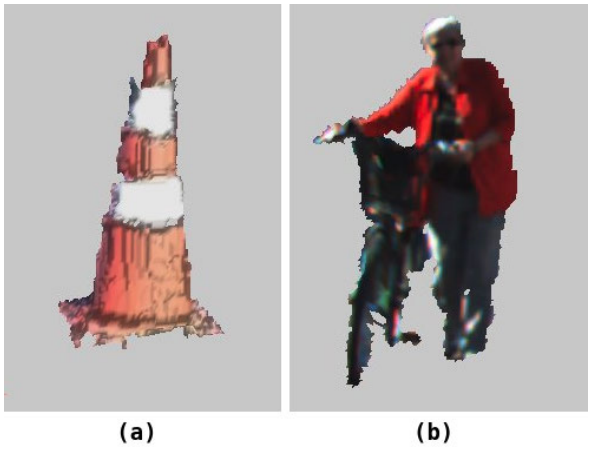


FIGURE 13. 3D-segmented objects (a) Cone - traffic sign, (b) Cyclist.

it possible to classify the pertinence of the detected traffic sign.

This work adopted an approach based on Multiple Attribute Decision-Making (MADM) for decision making, which was a class of arbitrary capable deals with the decision by evaluating a set of different criteria. According to Chen and Hwang [44], MADM applies very well to our problem, because it specializes in problems with finite sets of alternatives and enables evaluation in the decision steps [45].

2) ANALYTIC HIERARCHICAL

In the Analytic Hierarchical Process (AHP) technique, a hierarchical structure is created, thus making it possible to relate the components of the decision problem. With this feature of decomposition, the decision maker can make a comparison between the elements and classifies them into their priority level [45]. The step by step of this process can be followed in the work of Pachego and Bruno [45].

3) FUZZY REGIONS OF INTEREST: MULTIPLE ATTRIBUTE

A Fuzzy set is used with Multiple Making Attribute Decision Making (MADM) methods to model uncertainty and

subjectivity in decision analysis. Chen and Hwang [44] described some approaches to MADM steps. In this work we use fuzzy sets to represent the importance of each traffic sign detected in its regions of interest, thus, defining the priorities for each detected sign [45].

As we are working on the fuzzy linguistic model, the method will be applied by the two steps described below:

- *Step 1: Convert linguistic terms into fuzzy numbers*
  The approach we apply in this paper uses a finite set of linguistic terms that can be adjusted to best describe the nature of attributes, where, for example:

  U = *{very high, high to very high, high, fairly high, medium, fairly low, low, low to very low. }*.

  The first step of fuzzy logic identifies a conversion scale (from 0 to 1) that represents the subset used to characterize the attribute. By means of Figure 14 it is possible to observe an example of them. Since the fuzzy numbers on these scales can replace the linguistic terms [45].
- *Step 2: Convert fuzzy numbers into crisp score*
  Then, after identifying the corresponding scale of each value and replacing the linguistic terms with diffuse numbers, we have that a scoring method is applied to convert those numbers into sharp data. For this task, we propose a fuzzy scoring method for estimates μ_T, called *total score* of the fuzzy number M, using the left (μ_L) and right (μ_R) scores, the maximizing and minimizing sets (μ_{max} and μ_{min}, respectively) and the membership function of M (μ_M), where [45]:

$$\mu_L(M) = \sup_x[\mu_M(x) \wedge \mu_{max}(x)]$$
$$\mu_R(M) = \sup_x[\mu_M(x) \wedge \mu_{min}(x)]$$
$$\mu_T(M) = \frac{[\mu_R(M) + 1 - \mu_L(M)]}{2}$$

  Finally, replace each fuzzy number with the corresponding sharp score, the method results in an array with only sharp data, thus enabling the classic methods of MADM to classify the alternatives [45].

Through MADM it is possible to solve the decision problem using a M matrix, where the lines represent the alternatives, A, and the columns are the attributes, T, that evaluate the alternatives. The matrix M can be expressed as [45]:

$$M_{m,n} = \begin{pmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,n} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{m,1} & x_{m,2} & \cdots & x_{m,n} \end{pmatrix}$$

where we have: $A = \{a_1, a_2, \cdots, a_m\}$, $T = \{t_1, t_2, \cdots, t_n\}$, and $x_{i,j}$, with $i = 1, 2, \cdots, m$ and $j = 1, 2, \cdots, n$, are the values of each attribute.

4) TECHNIQUE FOR ORDER PREFERENCE BY SIMILARITY TO IDEAL SOLUTION

To rank the ideal solution, we use the algorithm based on Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) with the central concept that this approach is able to find the best alternative through the closest Euclidean distance to the ideal solution. The main steps of TOPSIS are shown below [44], [45]:

Algorithm 1 TOPSIS Algorithm

***Step 1**: Calculate the normalized decision matrix*

$$r_{i,j} = \frac{x_{i,j}}{\sqrt{\sum_{k=1}^{n} x_{k,j}^2}}, j = 1, \cdots, m$$

***Step 2**: Calculate the weighted normalized decision matrix*

$$v_{i,j} = w_j r_{i,j},$$

with $i = 1, 2, \cdots, m$ and $j = 1, 2, \cdots, n$.

***Step 3**: Determine the ideal and negative-ideal solutions*:

$$A^+ = \left\{ \left(\max_i v_{i,j} \mid j \in J \right), \left(\min_i v_{i,j} \mid j \in J' \right) \right\} \quad (1)$$

where $i = 1, 2, \cdots, m$ and J is the benefit attributes and J' is the cost attributes. For the decision problem, we tried to maximize J and minimize J'. The same is applied to the negative-ideal.

***Step 4**: Calculate the separation measures*

In this step the distance of each alternative from the ideal positive and negative solution must be calculated.

***Step 5**: Calculate the relative closeness to ideal solution*

Relative closeness is defined as:

$$c_{i^+} = \frac{s_{i^-}}{s_{i^+} + s_{i^-}}, 0 < c_{i^+} < 1, i = 1, 2, \cdots, m$$

***Step 6**: Rank the preference order*

Through the relative closeness c_{i^+} of each alternative is possible to rank them.

5) FUZZY KNOWLEDGE BASE: ATTRIBUTES THE SYSTEM

The input variables of the visual attention system are mapped to the following fuzzy sets:

6) REGIONS OF FUZZY INTEREST

In the graph of Figure 14 it is possible to observe the regions of fuzzy interest in their linguistic terms and which are generated to represent visual attention at their levels of importance and which are best visualized in Figure 15-(b).

By means of Figure 15-(a) it is possible to observe the trapezoid that is generated to define the navigation area (Equation 2). Every traffic sign detected within the trapezoid has higher priority than those outside. Thus, the closer

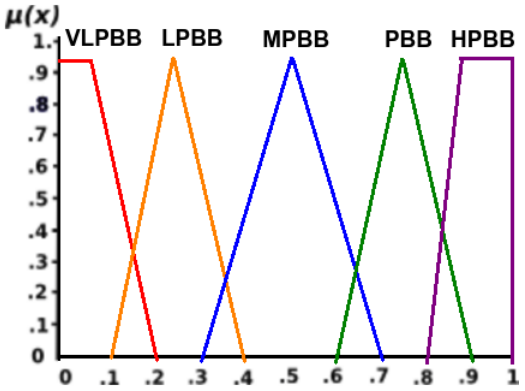


FIGURE 14. Regions of fuzzy interest - Legend:(HPBB = Hard Priority Bounding-Box; PBB = Priority Bounding-Box; MPBB = Medium Priority Bounding-Box; LPBB = Low Priority Bounding-Box; VLPBB = Very Low Priority Bounding-Box).

a traffic sign is to the edge of the highway, or outside it, the trapezoidal function should generate a smaller value (Figure 15-(b)).

$$\text{trap}(x; a, b, c, d) = \max(\min(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c}), 0) \quad (2)$$

7) FUZZY DISTANCES

Fuzzy distances: this feature is linked to the Euclidean distances of the traffic sign analyzed and the cluster of other transit objects of the scene (cones, vertical traffic signs, traffic lights) (Figure 16). The Euclidean distances of the objects are applied to an analysis related to the connection of the objects of the scene.

8) CONNECTIVITY FACTOR

The connectivity factor is linked to the degree of connection between major traffic signs and scene elements (cluster: cones, signs inside the highway, stop signs and people). A) Relation between the distances of the regions of interest and B) finally, of weight each connection that is given by the level of importance of each element in the scene in relation with traffic sign detected.

9) BASIS OF FUZZY RULES

The attributes were selected through the capabilities of the computer vision system (Image processing + machine learning for detection and classification) capabilities available in this work, as well as the knowledge and experience of the authors about robotic and human visual attention problems (Specialist Knowledge). The most common problems were listed for analysis by the fuzzy rules based machine learning algorithm - TOPSIS. The attributes are (Table 1).

The attributes are based on the important characteristics of a traffic sign to be evaluated. The alternatives define the choice of the traffic sign that is most relevant for the navigation of the vehicle in a certain section of the highway (Table 1).

TABLE 1. Attributes and alternatives for the diagnostic analysis.

Alternatives	Definition
a_1	Select traffic sign A
a_2	Select traffic sign B
a_3	Select traffic sign C
Attributes	**Definition**
t_1	Detect emergency traffic signs
t_2	Traffic sign on road
t_3	Detection of person near the traffic sign
t_4	sign connected with emergency elements
t_5	Increase speed
t_6	Distance to cluster of emergency signs

TABLE 2. Linguistic terms to diagnostic attributes values.

Attributes	Status	Linguistic Term		
		a_1	a_2	a_3
t_1	YES	very good	good	very bad
	NO	very bad	good	very good
t_2	YES	very good	good	very bad
	NO	very bad	good	very good
t_3	YES	very bad	bad	very good
	NO	very good	good	very bad
t_4	YES	very good	good	very bad
	NO	bad	good	very good

Through a based on Fuzzy Inference Systems auxiliary layer between the detection and decision-making system it was possible to evaluate the priority of each traffic sign for navigation of the vehicle. For this to be possible, a fuzzy rule base has been developed to support the inference system. The visual attention system was able to handle the proposed problem situations.

F. SEMANTIC SEGMENTATION: AN OVERVIEW

Semantic segmentation can be defined as a dense classification problem, where each pixel in the input image is associated with a given label from the set of classes under consideration. Initially, the task was addressed from a hand-crafted perspective, where feature extraction through image processing techniques - such as SIFT and HOG descriptors [46], [47] - and classification were performed in separate steps. Nonetheless, besides involving several processing steps for feature extraction and classification, hand-crafted methods were not robust to different situations, delivering the best performances in scenarios they were tuned for.

It was in 2015 that a major breakthrough took place: the proposition of Fully Convolutional Networks (FCNs) [48], which first tackled semantic segmentation as a dense classification problem by leveraging Deep Learning. Based on Convolutional Neural Networks (CNNs), FCNs made it possible to automatically and robustly perform end-to-end feature extraction and classification for images at any input resolution. Based on the advantages offered by FCNs, and contemporary works [49], [50], the literature on Deep Learning-based Semantic Segmentation (Deep Semantic Segmentation, or DSS) witnessed major improvements in robustness and precision, originating an accuracy-oriented research line.

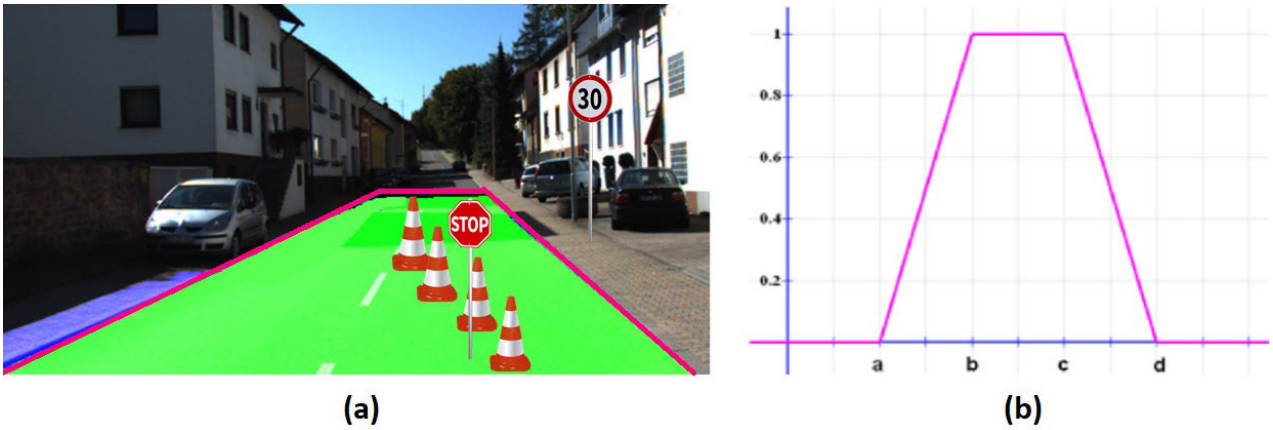


FIGURE 15. Region trapezoidal of fuzzy interest (a) Detection of highways and (b) Corresponding trapezoidal function.

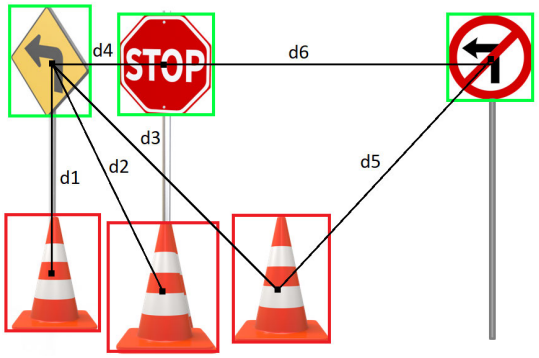


FIGURE 16. Calculation of the Euclidean distances between the traffic signs.

Among the main strategies employed for accuracy improvement, stand out: complex feature fusion schemes [51], local and global context modeling by means of dilated [52] and deformable convolutions [53], feature pyramids [54] and attention [55], as well as alternative approaches, such as spatial-aware operations for data-driven behavior [56], [57], [58].

Although current accuracy-oriented models produce very consistent results in terms of precision, reaching almost 87% accuracy in urban scene segmentation tasks,[2] this comes at the cost of high computational requirements. In addition to that, current literature on DSS is still limited in terms of 3D and temporal reasoning capabilities. An outline of the following sections is presented in Fig. 17. Table 3 presents a compilation of the main characteristics of the DSS methods covered in this review.

1) EFFICIENCY-ORIENTED SEMANTIC SEGMENTATION

The development of efficient models is of utmost importance in systems where multiple processes run in parallel, or where a full stack of processes must complete execution in real-time, under a limited set of hardware resources.

ADAS and autonomous vehicles are good examples of such systems. In first place, perception setups are composed by a diverse set of sensors, which operate in parallel for a robust understanding of the environment. Besides that, considering that perception is the first step in a more complex stack of modules for autonomous navigation, it should be performed efficiently and as quick as possible, then propagating its results to subsequent localization, mapping, planning and actuation modules. Current accuracy-oriented DSS models, however, do not match these requirements, thus finding limited application in such scenarios.

As a result of that, a recent trend in DSS research concerns the reduction of computational costs by means of the design of efficient strategies, which can be divided into input-level, architecture-level, and operation-level techniques.

Input-level techniques involve reducing the resolution [56] or cropping the input images [59], [60], [61], so that to limit the computational costs of model inference. However, it may lead to loss of spatial details, mainly related to small objects.

Architecture-level techniques propose to save resources either by adopting lightweight backbones, sharing weights and layers, reducing and reusing features, employing knowledge distillation, or designing modules and models from scratch.

Asymmetric encoder-decoder architectures follow the intuition that decoders can be kept smaller than encoders, thus saving computations [62], [63], [64], [65]. Similarly, multi-branch asymmetric methods apply separate asymmetric encoder branches for context and detail extraction [66], [67], [68]. Another common technique concerns the construction of models on top of lightweight pre-trained backbones [59], [63], [65], [69], [70], [71], such as ResNet-18 [72] and MobileNet [73], [74]. Although allowing to leverage pre-trained weights, inserting new elements can be challenging, and fine-tuning is required for dealing with domain shift.

[2]Benchmark Suite - Cityscapes Dataset. Available: https://www.cityscapes-dataset.com/benchmarks/

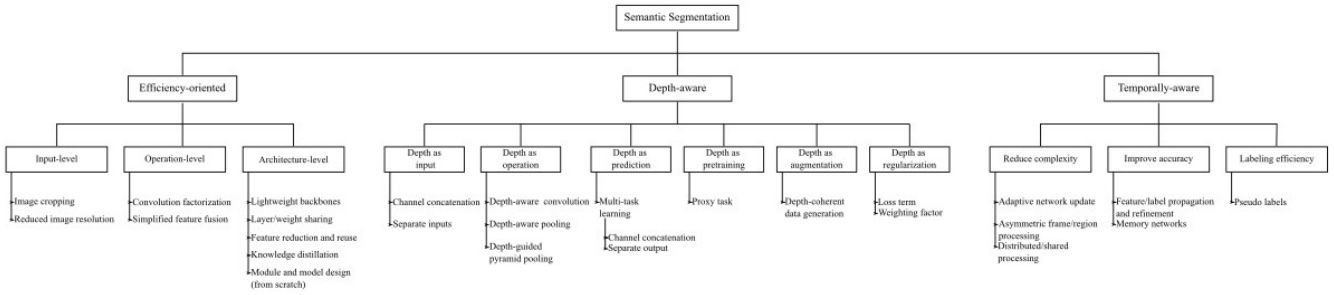


FIGURE 17. Outline of the discussion on efficiency-oriented, depth-aware and temporally-aware Deep Semantic Segmentation methods applied to urban scenes.

Layer and weight sharing reduce model complexity by sharing encoding layers. In Faster BiseNet [75], STDC Net [76] and EACNet [77] the first encoding layers are shared; afterwards, the extracted features follow to separate detail and semantic branches. In Fast-SCNN [78], a ''learning to downsample'' module performs shared low-level feature extraction to feed multi-resolution branches.

Feature reuse is usually leveraged in video semantic segmentation frameworks [79], [80], [81], where redundancy between adjacent frames can be used for feature and label propagation, hence skipping considerable processing steps.

Reducing feature size is also an alternative, generally achieved by fast down-sampling [82] or by reducing feature map dimensions [68], [70], [83], [84]. The main drawback concerns reduced learning capability: feature size reduction leads to loss of spatial information, while channel pruning harms modeling ability.

Knowledge distillation describes the process where a lightweight student network tries to mimic the behavior of a larger teacher model, with minimal loss in accuracy. It can be done by aligning the models' outputs [85], [86] or both outputs and some of their intermediate stages [87], [88].

Finally, other approaches rely on designing modules and models from scratch [64], [89], according to the application's needs. Neural Architecture Search (NAS) has also been used for automatic efficient model design [90], [91]. One major drawback of hand-crafted models and NAS is the need for training the model from scratch, missing a huge regularization opportunity offered by knowledge transfer from larger and more diverse recognition datasets [63].

Operation-level strategies build or remodel operations, so to reduce computational requirements. Convolution factorization and simplified feature fusion schemes are some examples.

Convolution factorization, such as Depthwise Separable Convolutions [92] and Asymmetric Convolutions [93], [94], [95], works by rearranging convolution operations, resulting in considerably less computations. In ERFNet [93] and FASSD-Net [95], plain convolutions are replaced by sequences of 1D convolutions. EACNet [77] employs depthwise asymmetric convolutions and dilated convolutions, while ESPNet [96] designs a sequence of point-wise convolutions and spatial pyramid of dilated convolutions.

Simplified fusion operations, such as element-wise addition [62], [63] and channel-wise concatenation [59], [77], are also common practices. More sophisticated gating and attention mechanisms, despite yielding better results, incur more computations and memory consumption [61]. This problem can be tackled by changing the order of operations [70], reducing the dimensions of attention maps [87] and intermediate representations (key, query and value) [97], as well as by limiting the search regions [98], [99] - although reducing complexity, the introduction of priors in the form of limited search regions may harm model's flexibility.

The aforementioned efforts led to significant improvements in model efficiency, with some architectures achieving inference rates above 100 FPS [51], [59], [68], [77], [89], [90], [95]. Nonetheless, efficiency comes at the cost of reduced learning capabilities, making it difficult for efficient models to match the precision of their accuracy-oriented counterparts. In this context, depth and temporal reasoning are promising alternatives for accuracy improvement, while preserving model efficiency.

2) DEPTH-AWARE SEMANTIC SEGMENTATION

Before the popularization of 3D sensors, 2D data fostered important advances in Computer Vision; in fact, the majority of the literature on Deep Semantic Segmentation of urban scenes is based on 2D perception. Nonetheless, relying solely on 2D data may limit perception in complex scenes. For instance, regions with no clear distinctions in the RGB space, although belonging to different classes, can be easily identified in depth maps [100] (Fig. 18). Depth also gives the model additional geometric cues about the scene, allowing it to distinguish between illustrations and real elements (Fig. 19), as well as to more robustly model elements with strong geometric priors, such as road, sidewalk and walls [101], reducing issues related to inconsistent and incomplete segmentations (Fig. 20). Finally, in scenarios of data scarcity, the auxiliary supervision offered by depth in multi-task learning setups can be useful for extracting complementary information for the target semantic segmentation task.

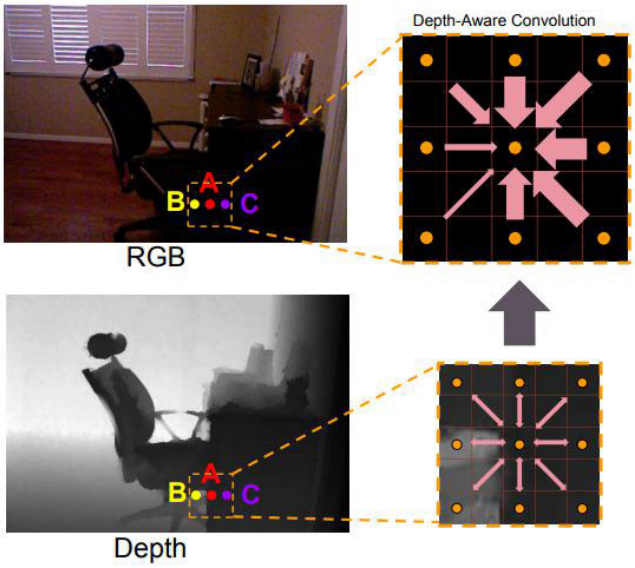


FIGURE 18. Regions without clear distinction in the RGB space can be identified in the associated depth map [100].

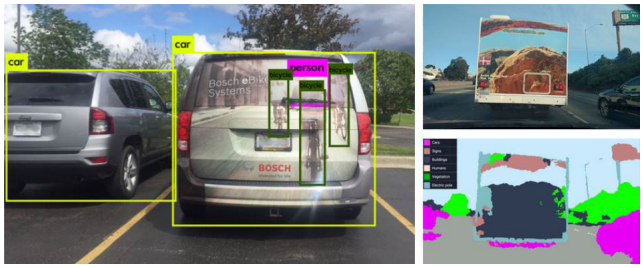


FIGURE 19. Examples of errors in 2D perception [102].

The use of depth information in depth-aware DSS models can be categorized into: depth-as-input, depth-as-operation, depth-as-prediction, depth-as-pretraining, depth-as-augmentation, and depth-as-regularization.

Depth-as-input strategies treat depth information as an additional input to the network, either in the form of an additional channel [103], [104] to RGB images, or processed as a separate input by a dedicated branch [105], [106] - some works even feed the stereo pair to the network [107], [108], [109]. Input fusion, although being faster, yields limited performance, while multi-branch approaches can get computationally expensive, due to the use of modality-specific encoders.

In depth-as-operation, depth is embedded into typical CNN operations, such as convolution and pooling, so to guide their behavior. Depth-aware convolution and pooling operations are proposed in [100], where neighboring pixels are weighted according to their depth similarity to a central pixel. The Spatial Information guided Convolution (S-Conv) [56] employs depth do adapt both the receptive field and convolutional weights. A depth-guided pyramid pooling has also been proposed in recent literature [110], [111]. In summary, depth-as-operation allows to directly embed depth into the network at a relatively cheap computational cost. Nonetheless, it still finds limited application, one of the main reasons being the non-fixed, and possibly deformed, receptive field generated when depth is used as a modifier for sampling locations, which leads to sub-optimal suitability for current accelerators, when compared to plain convolutions.

Depth-as-prediction usually describes multi-task learning setups, where depth is employed as source of auxiliary supervision for the extraction of complementary features, leading to a more robust perception. In fact, some authors support that depth estimation and semantic segmentation are correlated in terms of sample difficulty [112]. According to [113] domain-robust correlations between semantics and depth – e.g., sky is always far away, while roads are always flat – have the potential to largely improve the target semantic segmentation performance in the presence of a domain shift. Depth can be predicted as an additional channel to the segmentation output [114], or as a separate output from an auxiliary head [115], [116]. One of the main advantages of leveraging depth as prediction is that it does not require depth sensors during inference, what allows the use of cheaper camera sensors - nonetheless, in most cases, depth is required for supervised training. One intrinsic limitation is that multi-task learning can lead to very complex and computationally heavy setups.

Depth-as-pretraining employs depth as a proxy task for weight initialization, followed by fine-tuning on semantic segmentation (target task) [117]. Some approaches dealing with domain adaptation and cross-domain learning propose a similar method [101], [118], [119]. One of the main advantages of this approach is that it allows depth to be embedded into the network without the need for side structures, such as task-specific encoders and decoders. Besides that, domain shift issues can be tackled by employing data from the target domain during pre-training, so that to start the fine-tuning process with weights from the same domain as the target task.

Depth-as-augmentation aims to generate depth-coherent new (pseudo) samples out of existing labeled data, usually through blending mechanisms [101], [112].

Finally, depth-as-regularization explores depth information as an additional term or a weighting factor in the loss function. It can be used to penalize inconsistent boundaries between semantic and depth predictions [120], or to give more importance to closer objects in the per-pixel cross-entropy loss [121]. The main advantages of this technique are (i) the possibility of explicitly focus on certain ranges, according to the requirements of the task at hand, and (ii) the increase in accuracy without additional structures. Nonetheless, it explicitly adds inductive bias, thus reducing flexibility – in [121], for instance, model attention is directed to closer elements.

As a final remark, the majority of the literature on Deep Learning-based 3D perception concerns indoor scenarios,

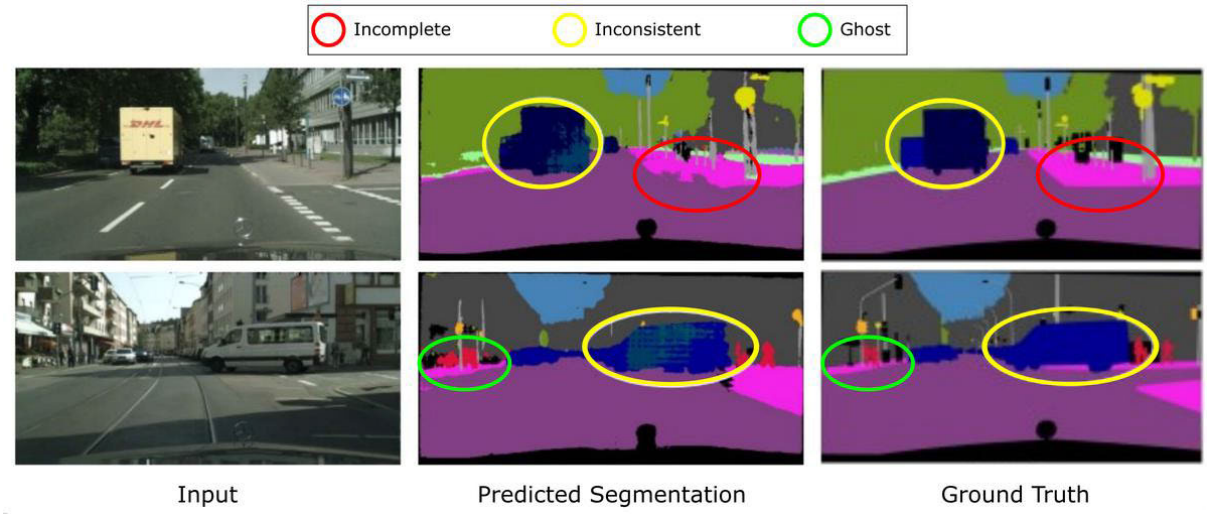


FIGURE 20. Inconsistent segmentations inside large classes (yellow), ghost (green) and incomplete (red) segmentations are some of the common errors in Deep Semantic Segmentation. Adapted from [82].

where 3D perception can be performed from structured light,[3] and depth ranges are limited. In urban scenarios, though, the use of structured light is not possible, and greater depth ranges are preferred, specially for critical applications, such as ADAS and autonomous vehicles. Hence, stereo vision is employed, allowing the dense perception of 3D structure, and at a fraction of the cost of other sensors - LiDAR, for instance. Nonetheless, its use for Semantic Segmentation is relatively under-explored, with works mainly relying on Multi-task Learning (MTL) setups. Some of the reasons are: limited range - compared to LiDAR -, noisy readings for higher depths, and failures related to reflexive and low-texture regions. Therefore, there is still much room for improvements in this research area.

3) TEMPORALLY-AWARE SEMANTIC SEGMENTATION

Although delivering more robust results, RGB-Depth Semantic Segmentation lets untouched another important factor to video DSS: temporal reasoning. Great part of current methods treats input frames independently. This neglects the possibility of exploiting temporal correlations in order to leverage redundancy, coherence and motion as additional cues to improve accuracy and reduce computations. Current temporally-aware DSS models can be classified according to their main goal, which can be: reduce computational costs, improve accuracy, or reduce the need for labeled data.

The intuition that semantic information changes at a slower pace are what allows us to explore redundancy between nearby frames for reducing the computational burden in video semantic segmentation. Works under this category are usually characterized by adaptive processing schedules. Clockwork [122] proposes a network update mechanism, where deeper layers, with stronger semantics, are updated less frequently than shallower ones. Another strategy widely adopted is asymmetric frame processing, where a video is divided into key and non-key frames. While for key frames low and high-level features are extracted, non-key frames pass through lighter architectures where only low-level features are computed; high-level features, in turn, are obtained by the propagation and aggregation of high-level features from the key frame [79]. In Accel [123], key frames are processed by a reference branch, while non-key frames are processed by a smaller update branch (Fig. 21). Finer granularity is studied in DVSNet [124], where, instead of asymmetric frame processing, the authors propose asymmetric region processing. It's worth noting that some authors argue that such an unbalanced processing scheme is harmful to model performance [125]. Distributed/shared processing is also a strategy for reducing computations. TDNet [87] distributes the computation of high-level features to several low-level feature computations from previous frames, performed by shallower networks. Temporal consistency can also be used to guide knowledge distillation mechanisms [126].

Improving accuracy in videos is usually related to also improving temporal stability, so that to build perception models that are able to maintain consistency through time by the aggregation of past information, as well as to robustly adapt to sudden changes, based on current data. Temporal instability (Fig. 22) refers to the presence of considerable fluctuations between consecutive predictions. This is particularly dangerous in autonomous navigation, since it can lead to the loss of previously identified objects, such as vulnerable road users (Fig. 22a), as well as the confusion between safe and unsafe regions through time, such as road and sidewalk (Fig. 22b). According to Fig. 22, both short and long-term temporal

[3]Kinect for Windows - Windows apps | Microsoft Learn. Available at: https://bit.ly/3T44Hs2

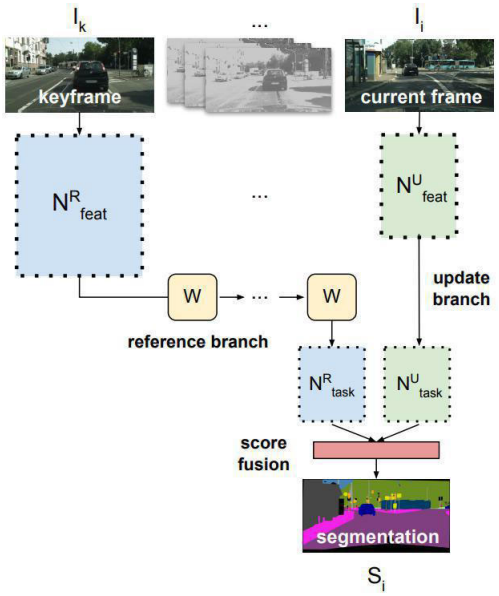


FIGURE 21. Accel architecture [123]. The authors propose a reference branch and a smaller update branch in order to process key and non-key frames, respectively.

instability may occur. Methods for accuracy improvement can be divided into: feature/label propagation and refinement, and memory networks. The first family of methods employs propagation of features and/or labels to neighboring frames, and then corrects discrepancies in the warped representations, usually based on motion cues [79], mismatching and uncertainty [127], [128], or previous features [123], [129]; afterward, these corrected representations are aggregated with the representations extracted for the frame under consideration, for feature refinement. In Memory Networks, features computed from previous frames are stored using a memory or backup mechanism, and subsequently recovered to enhance the segmentation of the current frame, which is used as a query to retrieve highly-correlated information in attention-based memory read [70], [83], [98], [99].

Finally, methods for increasing labeling efficiency address the problem of label scarcity in semantic segmentation scenarios. Due to the costs involved in data labeling, the most adopted video datasets for autonomous driving, such as Cityscapes and KITTI [130], do not have labels for all their frames. Instead, they provide labels for singular frames in a video sequence – e.g., Cityscapes only provides labels for the 20^{th} frame of its 30-frame video snippets. In this scenario, various works try to leverage unlabeled frames and previous knowledge to overcome label scarcity, with pseudo-label generation being one the most common approaches in the recent literature. Some authors demonstrate that training segmentation models on datasets augmented by synthesized samples leads to significant improvements in accuracy [60]. Temporal consistency with respect to neighboring unlabeled frames is explored in [126] and [131]. In [61] and [128], pseudo-labels are employed as auxiliary source of supervision.

G. SEMANTIC SEGMENTATION APPLIED TO ADAS AND AUTONOMOUS VEHICLES

Semantic Segmentation has been widely adopted in the work development by the Mobile Robotics Lab (LRM). Particularly, semantic segmentation is greatly explored for navigable and non-navigable region segmentation. Aligned with the history of the field, initial contributions were based on separate steps for feature extraction and classification. With the advent of Deep Learning and Fully Convolutional Networks, end-to-end segmentation was explored. The concern with efficiency and the use of depth information are fundamental characteristics of works developed in the lab, and have been present since the first contribution was proposed. The use of temporal information, on the other hand, still finds limited adoption in our research on semantic segmentation for visual perception/navigation. Finally, in recent works, semantic segmentation has been explored as auxiliary task for dealing with domain shift issues and improving learning ability in hybrid supervised/reinforcement learning setups. A summary of the main characteristics of the contributions on Semantic Segmentation made by the LRM is shown in table 5.

1) SAFE AND UNSAFE REGION DETECTION

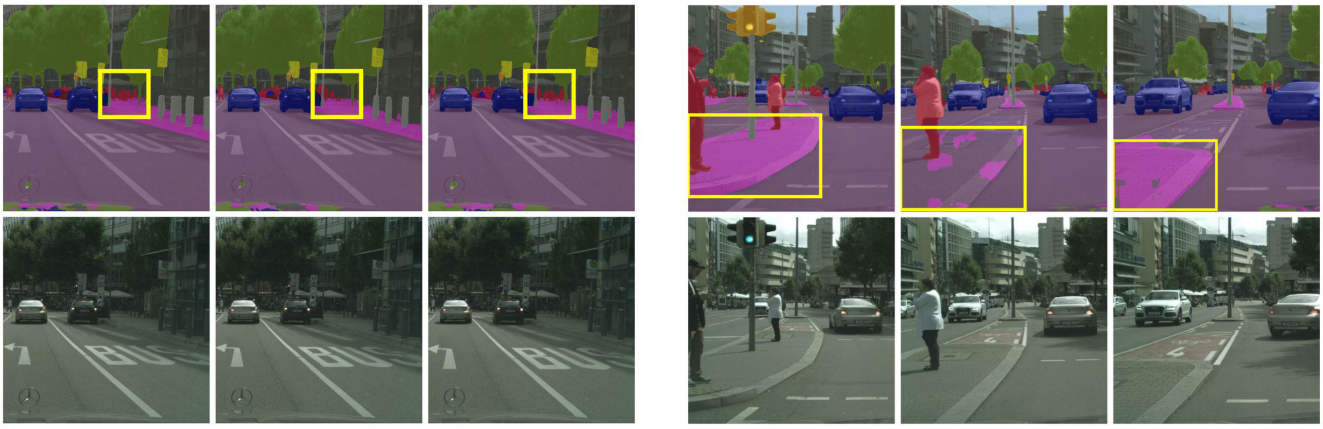
For vehicles to be able to safely navigate in outdoor environments, the detection of safe and unsafe regions is of utmost importance. In this respect, semantic segmentation plays critical role in segmenting the road from the input image. Initial works explored block-based classification methods, where the image is divided into square blocks. For each block, features - statistical measurements such as mean, probability, entropy and variance - are extracted manually, considering different color spaces (RGB, HSV and YCbCr). Afterwards, an artificial neural network is used for the classification of such elements into navigable and non-navigable regions [159], [160], [161]. The subdivision of the image into blocks also improves efficiency, allowing the method to operate in real-time.

Shinzato et al. [162] additionally apply a horizon identifier so that to only process the region below horizon line, and consequently improve system performance. The idea of only processing a Region of Interest (RoI) is also explored in [163]. Dias and Osório [164], implement and analyze a fixed-point Neural Network Ensemble for image segmentation applied to visual navigation, in order to improve efficiency.

Souza et al. [165] propose a vision-based navigation system, composed of navigable region segmentation, template matching to identify the geometry of the road ahead, a finite state machine to filter some input noise and reduce classification and/or control errors, and a template memory algorithm, which, based on an ANN and memory of templates from previous steps, generates steer angle and speed for vehicle control. This was the first example of temporal reasoning in one of our works. A detailed description of the system is depicted in Fig. 23. A similar approach is adopted in [166] and [167],

TABLE 3. Main characteristics of the Semantic Segmentation methods covered in our literature review.

Reference	Year	Data			Depth-aware strategy	Temporal reasoning strategy	Efficiency-oriented design		Goal
		2D	3D	Temporal			Strategy	Description	
[105]	2016	✓	✓		Depth as input				Semantic Segmentation
[122]	2016	✓		✓		Adaptive updating schedule	Efficient model design	Adaptive updating schedules	Semantic Segmentation
[62]	2017	✓					Reduced input resolution, Efficient model design	Single-stream Asymmetric Encoder-Decoder	Semantic segmentation
[66]	2018	✓					Efficient model design	Multi-branch asymmetric encoder	Semantic Segmentation
[68]	2018	✓					Efficient model design	Multi-branch asymmetric encoder	Semantic Segmentation
[81]	2018	✓		✓		Feature/label propagation	Asymmetric frame processing	Feature/label reuse	Semantic Segmentation
[93]	2018	✓					Efficient model design	Factorized convolutions	Semantic Segmentation
[96]	2018	✓					Efficient model design	Factorized convolutions	Semantic Segmentation
[100]	2018	✓	✓		Depth as operation				Semantic Segmentation
[103]	2018	✓	✓		Depth as input				Semantic Segmentation
[104]	2018	✓	✓		Depth as input				Semantic Segmentation
[110]	2018	✓	✓		Depth as operation				Semantic Segmentation
[111]	2018	✓	✓		Depth as Prediction, Depth as operation				Semantic Segmentation
[116]	2018	✓	✓		Depth as prediction				Multi-task learning (Semantic segmentation and depth estimation)
[117]	2018	✓	✓		Depth as pre-training				Semantic Segmentation
[124]	2018	✓		✓		Feature/label propagation and refinement	Asymmetric frame processing		Semantic Segmentation
[59]	2019	✓					Lightweight backbone		Semantic segmentation
[60]	2019	✓		✓		Feature/label propagation	Input cropping		Semantic segmentation
[65]	2019	✓					Lightweight backbone, Efficient model design		Semantic Segmentation
[78]	2019	✓					Efficient model design	Shared computations	Semantic Segmentation
[80]	2019	✓		✓		Feature/label propagation and refinement			Semantic Segmentation
[84]	2019	✓					Efficient model design	Reduced channel depth and shared convolutional weights	Semantic Segmentation
[97]	2019	✓		✓		Memory	Efficient model design	Sampling strategy for dimensionality reduction of attention maps	Semantic Segmentation
[108]	2019	✓	✓		Depth as prediction				Depth prediction (Semantic Segmentation as auxiliary task)
[109]	2019	✓	✓	✓	Depth as input	Feature/label refinement (LSTMs)			Semantic Segmentation
[118]	2019	✓	✓		Depth as pre-training				Domain adaptation (semantic segmentation as target task)
[121]	2019	✓	✓		Depth as regularizer				Semantic Segmentation
[123]	2019	✓		✓		Feature/label propagation and refinement	Asymmetric frame processing		Semantic Segmentation
[63]	2020	✓					Lightweight backbone, Efficient model design	Asymmetric Encoder-Decoder	Semantic Segmentation
[64]	2020	✓					Efficient model design	Single-stream network	Semantic Segmentation
[69]	2020	✓					Lightweight backbone, EffIcient model design	Factorized convolutions	Semantic Segmentation
[70]	2020	✓		✓			Lightweight backbone, Efficient model design	Adapted self-attention	Semantic Segmentation
[71]	2020	✓					Lightweight backbone, Efficient model design		Semantic Segmentation
[79]	2020	✓		✓		Feature/label propagation	Efficient model design, Assymetric frame processing	Hybrid CPU-GPU processing, Feature/label reuse	Semantic Segmentation
[85]	2020	✓					Knowledge distillation		Semantic Segmentation
[87]	2020	✓		✓		Distributed/parallel frame processing	Lightweight backbone, Efficient model design	Reduced encoders, Subsampled attention maps	Semantic Segmentation
[88]	2020	✓		✓		Feature/label propagation and refinement	Asymmetric frame processing		Semantic Segmentation
[91]	2020	✓					Efficient model design	Neural Architecture Search	Semantic Segmentation
[95]	2020	✓					Efficient model design	Factorized convolutions	Semantic Segmentation
[126]	2020	✓		✓		Feature/label propagation, temporal consistency loss, temporally-guided knowledge distillation	Knowledge distillation		Semantic Segmentation
[128]	2020	✓		✓		Feature/frame propagation and correction	Efficient model design		Semantic Segmentation
[129]	2020	✓		✓		Feature/label refinement	Efficient model design		Semantic Segmentation
[56]	2021	✓	✓		Depth as operation				Semantic segmentation
[61]	2021	✓		✓		Pseudo-labels (from unlabeled neighboring frames)			Semantic segmentation
[67]	2021	✓					Efficient model design		Semantic Segmentation
[75]	2021	✓					Efficient model design		Semantic Segmentation
[76]	2021	✓					Efficient model design		Semantic Segmentation
[77]	2021	✓					Efficient model design	Shared computations, asymmetric convolutions and simple fusion	Semantic Segmentation
[83]	2021	✓		✓		Memory	Efficient model design	Shared backbone	Semantic Segmentation
[89]	2021	✓					Efficient model design		Semantic Segmentation
[90]	2021	✓					Efficient model design	Neural Architecture Search	Semantic Segmentation
[94]	2021	✓					Efficient model design	Factorized convolutions	Semantic Segmentation
[98]	2021	✓		✓		Memory	Efficient model design	Sampling strategy for dimensionality reduction of attention maps (local neighborhood search window)	Semantic Segmentation
[99]	2021	✓		✓		Memory	Efficient model design	Reduced attention map (key and query selection)	Semantic Segmentation
[106]	2021	✓	✓		Depth as input				Semantic Segmentation
[107]	2021	✓	✓		Depth as prediction				Semantic Segmentation
[112]	2021	✓	✓		Depth as Pretraining, Depth as Prediction, Depth as Augmentation	Consecutive frame processing for pose and motion estimation			Semantic Segmentation
[113]	2021	✓	✓		Depth as prediction, Depth as regularizer				Domain adaptation (semantic segmentation as target task)
[114]	2021	✓	✓		Depth as prediction		Lightweight backbone, Efficient model design	Single-stream network	Multi-task learning (Semantic segmentation and Depth estimation)
[115]	2021	✓	✓		Depth as prediction		Efficient model design		Multi-task learning (Semantic segmentation and depth estimation)
[119]	2021	✓	✓		Depth as pre-training		Lightweight backbone		Domain adaptation (semantic segmentation as target task)
[120]	2021	✓	✓		Depth as regularizer				Semantic Segmentation
[127]	2021	✓		✓		Feature/label propagation and refinement	Lightweight backbone		Semantic Segmentation
[131]	2021	✓		✓		Unsupervised temporal consistency loss			Semantic Segmentation
[82]	2022	✓					Efficient model design	Fast downsampling and factorized convolutions	Semantic Segmentation
[101]	2022	✓	✓		Depth as augmentation				Domain adaptation (semantic segmentation as target task)



(a) (b)

FIGURE 22. Single frame-based models may result in short-term (a) or long-term instability (b).

TABLE 4. Characteristics of the main datasets used for visual perception in autonomous vehicles.

Dataset	Year	Sensing Modality		Adverse Conditions		Objective		Size [frames]*	Link
		2D	3D (stereo)	Lighting	Weather	Detection	Segmentation		
KITTI [130]	2012	✓	✓	✓		✓	✓	30,000 (detection) 800 (segmentation)	https://bit.ly/3CvGRNr
MS-COCO [132]	2015	✓				✓	✓	200,000	https://bit.ly/3jMNMdG
KAIST Multispectral Pedestrian [133]	2015	✓		✓		✓		95,328	https://bit.ly/3CwoFn4
JAAD [134]	2016	✓		✓	✓	✓		82,032	https://bit.ly/3mnwOo6
Tsinghua-Daimler Cyclist Detection Benchmark Dataset [135]	2016	✓				✓		14,674	https://bit.ly/3BrKSBg
Playing for Data: Ground Truth from Computer Games [136]	2016	✓		✓	✓		✓	25,000	https://bit.ly/3q4Cwh5
SYNTHIA [137]	2016	✓	✓	✓	✓		✓	214,000	https://bit.ly/3jO3fdN
Cityscapes [138]	2016	✓				✓	✓	25,000	https://bit.ly/3pV2cN3
Oxford RobotCar [139]	2016	✓		✓	✓				https://bit.ly/3w0RlSK
Multi-spectral Object Detection dataset [140]	2017	✓		✓		✓		7,512	https://bit.ly/3nrMXYM
Multi-spectral Semantic Segmentation dataset [141]	2017	✓		✓			✓	1,569	https://bit.ly/3nrMXYM
Mapillary Vistas [142]	2017	✓		✓	✓		✓	25,000	https://bit.ly/3jPdxdk
KAIST [143]	2018	✓	✓	✓		✓		7,512	https://bit.ly/3pPw7G9
ApolloScape [144]	2018	✓	✓	✓	✓		✓	146,997	https://bit.ly/3nJIhO2
nuScenes [145]	2019	✓		✓	✓	✓	✓	40,000 93,000 (segmentation - nuImages)	https://bit.ly/3nDJgzC
SeeingThroughFog [146]	2019	✓		✓	✓	✓		13,500	https://bit.ly/3jOT4Wp
BLVD [147]	2019	✓		✓		✓		120,000	https://bit.ly/3pXO7y2
Waymo Open Dataset [148]	2019	✓		✓	✓	✓		200,000	https://bit.ly/3w4uPIv
H3D [149]	2019	✓				✓		27,721	https://bit.ly/3nJXsH6
A2D2 [150]	2019	✓			✓	✓	✓	41,280 (segmentation) 12,499 (detection)	https://bit.ly/3pXQO2C
A*3D Dataset [151]	2019	✓		✓	✓	✓		39,000	https://bit.ly/3jNmnZc
EuroCity Persons [152]	2019	✓		✓	✓	✓		47,300	https://bit.ly/3pQKkmq
Argoverse [153]	2019	✓	✓	✓	✓	✓		113 recordings (15 to 30 seconds long) 50,850 frames (22.5 seconds long sequences, sampled at 20 Hz)	https://bit.ly/3GCFDSX
StreetHazards [154]	2019	✓		✓	✓		✓	7,656	https://bit.ly/3bm8wED
Brno Urban Dataset [155]	2019	✓		✓	✓	✓		67 recordings (summing up to 10h) 720,000 frames (frame rate of 20 Hz)	https://bit.ly/3GBoycg
Canadian Adverse Driving Conditions Dataset [156]	2020	✓			✓	✓		7,000	https://bit.ly/2XWcoJm
Berkeley Deep Drive (BDD100K) [157]	2020	✓		✓	✓	✓	✓	100,000 (detection) 20,000 (segmentation)	https://bit.ly/3jREwoL
RaDICaL [158]	2021	✓	✓			✓**	✓**	218,689	https://bit.ly/3BwT42Z

* When not explicitly mentioned otherwise. ** Although the authors mention the possibility of using the dataset for this purpose, no labeled data is provided.

where the environment is represented by a topological map in which each state is related to a specific track shape.

Semantic segmentation can also be used to identify lane markings, in order to localize and keep the vehicle on track. In [163], the authors only apply image processing techniques, not relying on any machine learning method. A filter is first applied to accumulate differences of intensity between pixels; then, Otsu thresholding is applied for image binarization. Finally, a Probabilistic Hough Transform is employed to detect all possible straight lines. Temporal data is used for

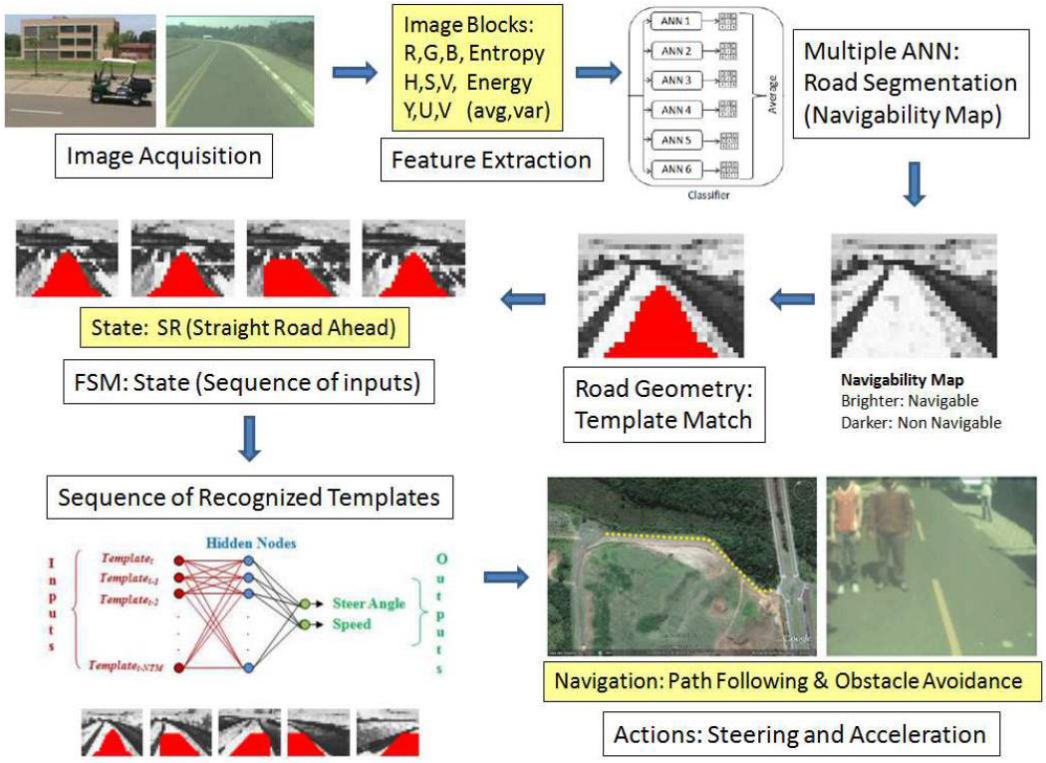


FIGURE 23. General outline of the vision-based autonomous navigation system proposed in [168].

road marker estimation, when the model is not able to detect them correctly.

Parallel to the aforementioned contributions, depth has also been explored in order to improve model robustness. In [169], the authors propose a method for obstacle and free space detection based on disparity maps and a series of processing steps based on graphs and a cost value considering a local neighborhood of points. Shinzato et. al [170] propose to use depth from stereo camera, along with other features based on color, to detect the road region by means of Machine Learning techniques. Camera-LiDAR sensor fusion is used in [171] for road and obstacle segmentation.

We also emphasize the importance of rich contextual information to a more precise perception of the environment. Mendes et al. [172] employ an ANN for classification of hand-crafted features. A given reference block is classified based on features extracted for itself, as well as for a given context covered by the termed contextual blocks - Fig. 24. Besides that, road blocks are selected as references for visual information related to roads, in order to give more cues for the classification model.

Similarly to previous works in the lab, the method in [173] explores patch-wise classification of the image. However, the expansion of the receptive field is explored in order to cover more contextual information for the classification of a central square in the patch. Aligned with the advances in the field, the authors now employ CNNs for image segmentation, instead of ANNs.

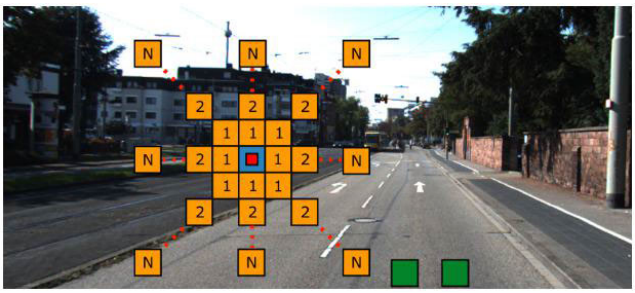


FIGURE 24. Illustration of the mechanism proposed in [172] for the processing of additional contextual information. The classification block is shown in red, the contextual blocks in orange, the possible support block in blue and the road blocks in green.

2) SEMANTIC SEGMENTATION IN MULTI-TASK LEARNING

One of the goals of semantic segmentation is to provide a summarized representation of the image, with a description of its elements in terms of classes, instead of the raw image information - color, texture, depth - acquired through vision sensors. Aligned with that, the contributions described in the previous section were mainly based on binary semantic segmentation, in which input images were translated into representations where only road and non-road classes were usually considered. Considering a greater number of classes, however, can bring more meaningful information about the environment, contributing to autonomous navigation. In this sense, recent works developed by our members

apply multi-class semantic segmentation instead of its binary version. Horita et al. [174], for instance, apply semantic segmentation as both proxy task for model pre-training, as well as auxiliary task in multi-task learning by leveraging both supervised and reinforcement learning. The authors prove that the proposed strategy leads to better robustness to noise and environmental adverse conditions, as well as allows faster convergence and lower variance during training.

3) FUTURE DIRECTIONS OF SEMANTIC SEGMENTATION

Semantic segmentation has proven to be essential in visual perception for autonomous navigation. In great part of the works developed in the Mobile Robotics Lab, semantic segmentation has played fundamental role in the identification of navigable regions, obstacles, and also as auxiliary task for increasing robustness and dealing with domain divergence issues related to transfer learning. Besides that, depth has been widely adopted as additional source of information from the environment, helping in increasing perception performance. Additionally, efficiency is of utmost importance to autonomous navigation systems, and, therefore, has been a major concern in our work, mainly in the form of hardware optimization and network simplification strategies.

Despite that, there is still much room for improvements concerning not only semantic segmentation, but visual perception in general. First, video sequences are the natural source of visual information acquired from cameras; nonetheless, although being critical to guaranteeing perception stability, the correlation and redundancy among frames in video sequences have not yet been explored to their full potential in our work. In second place, efficiency has been tackled mainly from the perspective of simplified inputs and hardware for image processing; adopting an efficiency-oriented line of design in our models seems to be the natural path for improving perception efficiency, according to the literature. Finally, depth has been used in our work as either pre-processing steps or additional inputs; in spite of that, employing depth as additional source of supervision in multi-task learning setups for improving model performance, as well as embedding depth into network operations so that to improve efficiency and depth-awareness with reduced costs, seem to be promising directions.

In summary, we believe that the future of semantic segmentation for visual navigation is intimately related to efficiency-oriented design, depth-aware models and operations, as well as temporal reasoning for model robustness and stability.

H. DRIVER MONITORING

The high cost of implementing fully autonomous vehicles with a high degree of reliability leads us to believe that autonomous vehicles will need to share the streets and roads with traditional vehicles driven by humans, at least for a relatively long period of time.

In this sense, an important step to minimize errors in traffic is the possibility of the vehicle monitoring the driver, so that to detect monitor, and analyze behaviors of vehicle drivers, making it possible to indicate their ability or inability to operate the vehicle.

This subsection deals with works related to driver monitoring and safe/unsafe driving verification. Among these stand out the detection of drunkenness (Seção III-H1), drowsiness (Seção III-H3) and the use of a cell phone by the driver (Section III-H3). In addition, vehicle driving evaluator and classifier systems are also relevant for driver monitoring (Seção III-H4).

1)

There are different methods for detecting a drunk driver. The drunk driver detection system developed by [175] and improved in [176] uses a path between sensors (arranged in parallel) through which the driver needs to drive the vehicle. A normal driver can stay in a safe/normal region within the path. The sensors are portable, hollow and must be positioned on the ground. When the vehicle's tire is placed on a sensor, its internal volume is reduced and signal lamps are activated, or even other devices, thus indicating the driver's ineptitude.

In the case of ignition control systems, one can mention the system proposed by [177] that checks the alcohol content in the driver's blood using a transdermal sensor incorporated into the steering wheel of the vehicle. Being above a threshold, the system prevents the vehicle from starting. The measurement of alcohol content is done by neural algorithms previously trained to receive information from sensors. Another proposal of this nature is that of [178] which works in a similar way, however, the alcohol content is measured by the intensity of the wavelengths emerging from the driver's finger. A microprocessor correlates the collected intensity with the alcohol content. To ensure that a finger of the vehicle's main driver is the one being analyzed, the system identifies the fingerprint.

The [179] system is based on a sensor for human breathing that detects whether the driver is drunk or not (above a permissible limit of alcohol in the breath). The sensor is installed and measures the alcohol concentration in the region close to the driver's seat.

Reference [21] presented a non-intrusive system that detects if a driver is under the influence of alcohol by measuring the driver's pulse. A window of 180 seconds is used to compare the acquired driver data with a database of normal and drunk people. Thus, the system is able to predict between a drunk driver and a normal one.

The solution presented by [180] focuses on the detection, using inertial sensors, of dangerous driving performed by a drunk driver. A program installed on the cell phone computes data from an accelerometer and an orientation sensor, then comparing it to a dataset with typical patterns of a drunk driver to identify improper driving. The system can alert the

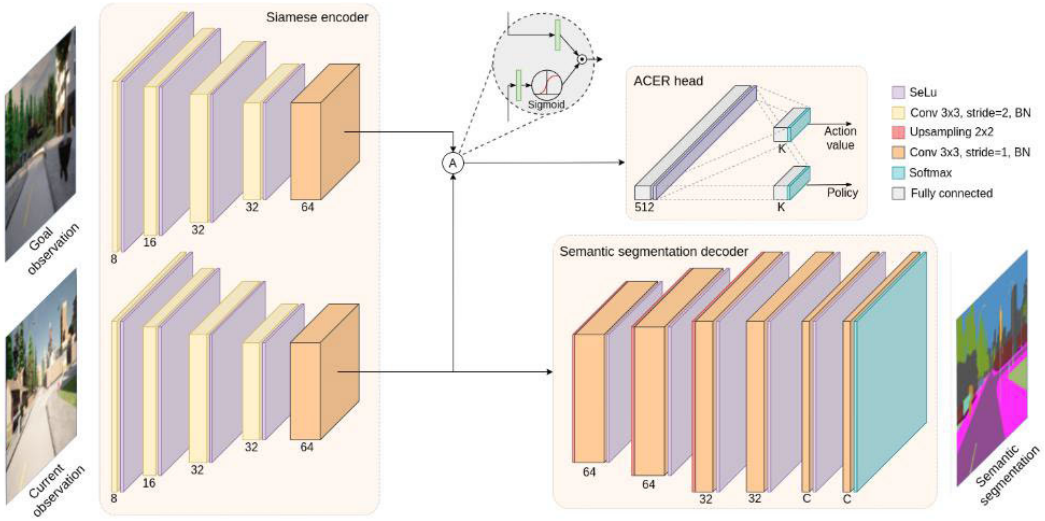


FIGURE 25. Multi-task learning architecture proposed in [174], for joint exploration of supervised (semantic segmentation) and reinforcement learning (action and policy).

TABLE 5. Main characteristics of the works on Semantic Segmentation from our Mobile Robotics Lab (LRM).

Method	Reference	Year	Data			Depth-aware strategy	Temporal reasoning strategy	Efficiency-oriented design		Goal
			2D	3D	Temporal			Strategy	Description	
Hand-crafted feature extraction and ANN-based classification	[159]	2011	✓					Reduced input resolution	Subdivision of the image into blocks, processed independently	Road segmentation
	[160], [161]	2011	✓					Reduced input resolution	Subdivision of the image into blocks, processed independently; RoI (below the horizon line)	Road segmentation
	[168]	2011	✓		✓		Memory (previous states)	Reduced input resolution	Subdivision of the image into blocks, processed independently	Visual navigation (road segmentation as intermediate task)
	[165]	2011	✓		✓		Memory (previous states)	Reduced input resolution	Subdivision of the image into blocks, processed independently; RoI (below the horizon line)	Visual navigation (road segmentation as intermediate task)
	[170]	2011	✓	✓		Depth as input		Reduced input resolution	Subdivision of the image into blocks, processed independently	Road Segmentation
	[162]	2012	✓					Reduced input resolution	Subdivision of the image into blocks, processed independently; RoI (below the horizon line)	Road segmentation
	[164]	2012	✓					Reduced input resolution, Efficient model design	Subdivision of the image into blocks, processed independently; Fixed-point network implementation (instead of floating-point)	Road segmentation
	[166], [167]	2012	✓					Reduced input resolution	Subdivision of the image into blocks, processed independently	Visual navigation (road segmentation as intermediate task)
	[163]	2014	✓					Input cropping	RoI (road in front of the vehicle)	Lane segmentation
	[172]	2015	✓					Reduced input resolution	Subdivision of the image into blocks, processed independently	Road Segmentation
Graph-based neighborhood weighting (object segmentation) and polar histograms (road segmentation)	[169]	2014	✓	✓		Depth as input		Reduced input resolution	Subdivision of the image into blocks, processed independently	Road and obstacle segmentation
	[171]	2014	✓	✓		Depth as input				Road and obstacle segmentation
Deep Learning	[173]	2016	✓							Road Segmentation
	[174]	2021	✓							Visual navigation (semantic segmentation as auxiliary task for Deep Reinforcement Learning)

local police automatically by phone call in places where there is cellular coverage.

Reference [181]'s proposal identifies whether a driver is drunk by the way he drives the vehicle on the roadway. The

system locates the center of the roadway and the curvature of the road, comparing them to the steering wheel angle over a period, it is possible to distinguish the behavior of a sober individual from an alcoholic one. The model of distinction between sober and drunk was obtained using a data set obtained by drivers in a simulator. The method generates personalized models (sober and drunk) for each driver, that is, the driver needs to be identified the moment he starts driving the vehicle. Thus, with individual models, the system achieved high fidelity. The method uses *online* techniques that update the drivers' models while driving. In the case of generating sober and drunk generic models (suitable for any driver), the method presents high uncertainty in its detection.

2) DROWSINESS DETECTION

Among the works in the area, the approach by [182] stands out, which uses a camera facing the driver. The proposal locates the face and then the eyes of the driver using the Viola-Jones [183] region detection technique. Using the Hough Circular Transform [184] in the eye region it is possible to identify the iris and the two eyelids. The state of the eyes is obtained by classifying two characteristics, one non-linear and the other non-stationary, thus allowing the detection of sleepiness.

Reference [185]'s work presents a system that extracts characteristics of ocular amplitude, such as the amplitude itself, energy, maximum and average speeds of closing and opening the eyes, and characteristics linked to duration, such as frequency of closing and opening, closed dwell time and the PERCLOS [186]. PERCLOS is the rate that the driver's eyes remain closed over a period of time (one of the most common characteristics in drowsiness detection systems), with PERCLOS greater than or equal to 80% being considered a risk. Using data from real and simulated experimentation with 43 individuals in a total of 67 hours, it was possible to reach an accuracy of 82% in detecting sleepiness. The real tests were carried out with 18 people who had their data collected using a route of 260 km on highways in Germany, during the daytime (between 9-13 hours) and with little traffic on the highways.

Reference [187]'s proposal detects drowsiness based on physiological signals (electrodes placed on the driver's head) and on the detection of eye closure time. The eyes are detected using Viola-Jones [183] starting from images acquired from a camera. In each frame it is classified whether the driver's eyes are open or closed. A *Fuzzy* technique is employed to analyze the physiological and closure data to detect sleepiness.

Another system is presented by [188], which performs driver segmentation by skin color and uses SURF (*Speeded Up Robust Feature*) [189] as a model for extracting facial features, used to determine whether the eye is closed and its rate of closure. The method uses these two pieces of information to detect drowsiness.

The algorithm proposed by [190] uses mathematical morphology in the segmentation of the driver's eyes, and the detection of the state of the eye, such as being open or closed, is possible with the aid of the Gabor filter (local spatial frequency distribution). Drowsiness detection occurs using PERCLOS.

In [191]'s work on sleepiness detection, the state of the eyes is identified by the presence or absence of the retina. Mathematical morphology is used to detect whether the eye is open or closed, and PERCLOS is used to detect drowsiness.

3) CELL PHONE DETECTION

There are active efforts to develop methods to detect mobile phone use by drivers while driving. The simplest way to solve this problem is to help the driver not to answer phone calls while driving. Usually this can be obtained using a specific mobile application, but in this case the driver needs to actively collaborate with the system for it to work. The ''Hands on the Wheel'' [192] system, developed by the Brazilian government, blocks any notification on a cell phone for a period of time, that is, while the driver is driving the vehicle. Calls are also automatically answered via text messages between cell phones, informing the person trying to contact the driver that he is momentarily unavailable as he is driving. As for receiving SMS, the app blocks message notifications. That is, if the driver receives a text message, he will not be notified, being able to view it as soon as he disables the application. However, whoever sent the SMS does not receive the notification that the person is driving. The start and duration of the block are indicated by the driver. The system alone does not prevent misuse of the cell phone.

Reference [193]'s approach uses *software* running on the cell phone to capture and process high-frequency sound signals sent by the vehicle's sound equipment. The sound signals are used to measure the position where the cell phone is, so it is possible to know if it is the driver who is using it, thus blocking the operation of the device. The proposal obtained a classification accuracy of more than 90% in detecting the driver, being experimented with two different models of cell phones and using two different vehicles. But this system depends on the mobile operating system and *software* needs to be continuously enabled. The great advantage of the technique is that it works even when using the cell phone with headphones (*hands-free*).

The [194] method uses *software* on the driver's cell phone and device sensors (accelerometer, gyroscope, compass and microphone) to detect events, the events make it possible to know if a driver is wanting to use the cell phone while driving. Examples of detected events are ''walking towards the vehicle'', ''standing close to the vehicle and opening one of the vehicle's doors'', ''entering the vehicle'', ''closing the door'' and '' starting the engine''. To reduce the chance of false detection of events, the method employs a state machine for the logical sequence of events. The position where the cell phone entered the vehicle is also identified, so the system is able to distinguish between a driver and a passenger. All activities are identified through the analysis and fusion of

the sensors, allowing the total blocking of the device if an ongoing driver distraction is detected.

The [195] method is an intrusive method and uses electroencephalography (physiological signals) to identify when the driver is talking on the cell phone or not. However, this system needs an initial configuration of the sensors and some electrodes (intrusive) distributed across the driver's chest.

The work done by [196] aims to achieve the detection and classification of car driver activities using a computer vision algorithm that detects relative motion based on the driver's skin segmentation. Another study using skin segmentation is presented by [197], where the driver's skin is detected through adaptive segmentation. From the binarized image of the driver's skin, two characteristics are extracted, the Hand Percentage (PM) and the moment of Inertia (first Moment of HU) [198]. The PM is the ratio of *pixels* of skin in two specific regions. Through these two characteristics, cell phone use can be predicted in each frame. If the presence of a cell phone is observed in at least 65% of the frames processed in a period of 3 seconds, the use of the cell phone by the driver (in the period) is declared by the system. The system is hybrid and also uses a system for detecting the movement of hands/arms (optical flow) as parameterization of the pattern recognition system. Both methods use a camera to acquire a sequence of images of the driver, but the first uses a driver's side camera while the second uses a driver's front camera. Image-based systems have some limitations due to problems with lighting and shadows.

The methods by [199] and [200] detect cell phone use using infrared surveillance cameras, in an attempt to avoid some problems with lighting. The systems locate the face of the driver (ROI). ROI is graded between cel use and non-use

4) UNSAFE DRIVING DETECTION

The system presented by [201] is capable of monitoring the driver using two image capture cameras, one positioned in front of the driver and the other in front of the vehicle. Using the internal camera, the system detects the direction in which the driver is looking. The direction of the road ahead is detected using the second (external) camera on the vehicle. These two extracted orientations are converted to the same coordinate system and correlated. Thus, the state of driving can be inferred between: safe, risky and very risky. A Hidden Markov Model (HMM) [202] analyzes the same correlation to detect 4 driving patterns, which are: straight track, curve in track, changing lanes and making a turn.

Reference [203]'s driving style detector system classifies the vehicle driver into typical (non-aggressive) and aggressive, using a *software* operating on a cell phone properly positioned on the car's windshield. The cell phone needs to contain a camera, an accelerometer, a gyroscope, a magnetometer and a GPS for the system to operate. The system detects whether or not the driver is aggressive by performing a non-linear mapping (*Dynamic Time Warping*) between the data currently acquired by the sensors and data from a small training set (models).

The DriveSafe application [204], available for iPhone-type cell phones, detects inattentive behavior and dangerous driving. The *software* even measures the driving quality and alerts you when this score is low. The definition of driving quality is based on detecting drowsiness and distractions. Drowsiness is detected through the way the driver drives the vehicle through the roadway (position and involuntary exit from the track), which is detected through the rear camera of the cell phone positioned to observe the roadways in front of the vehicle. The distraction detection uses the accelerometer, the gyroscope and the GPS, so it is possible to detect the levels of acceleration, braking and yaws made by the vehicle. The system does not operate in urban city traffic, only on highways, so it only works when the vehicle speed is greater than 50 km/h.

The system proposed by [205] diagnoses the driver's direction using a GPS to acquire the position, speed, acceleration and the turns made by the vehicle. These data are compared with previously cataloged rules for driving in the region, with rules based on law and safety techniques. A system based on Logic *Fuzzy* is responsible for qualifying the performed direction.

5) CONSIDERATIONS

This section presented works related to the detection of cell phone use in traffic, drowsiness, drunkenness and unsafe driving within the scope defined for this work. The systems raised in this section act in the monitoring of the driver, in the qualification of the direction or before the vehicle is started, thus allowing the detection of disturbances and distractions. The Table 6 shows the comparison between the systems presented in this section.

Some of the works raised [182], [187], [188], [190], [191], [196], [197], [200], [201], [206] that monitor the driver use RGB cameras for this purpose. These cameras rely on lighting and, moreover, a certain constancy and homogeneity of lighting to detect and target the driver correctly. The accuracy of driver segmentation can be impaired by the presence of interior parts of the vehicle with colors close to human skin. In a real situation, the incidence of lighting can vary, which makes this a very relevant problem. For example, in the parts where sunlight hits, the *pixels* of the acquired image saturate, that is, they tend to present the white color as can be seen in Figure 26. The movement of the vehicle also causes the displacement of the region of incidence of light. All these problems make it difficult to use a camera-based system to acquire driver data for a real environment.

A camera is capable of capturing 2D information from the scene, that is, we have no idea of *pixels* depth in relation to the camera. The use of a 3D Sensor that already uses its own lighting that is not visible and is tolerant of sunlight is interesting for this type of solution. Thus, 3D data (point cloud) can be used to track the movements and actions of the driver inside the vehicle, without any influence of color or

TABLE 6. Comparison table between related works and the developed system.

		[177]	[21]	[179]	[180]	[176]	[181]	[187]	[185]	[182]	[188]	[190]	[191]	[192]	[193]	[194]	[199]	[200]	[206]	[195]	[196]	[197]	[201]	[203]	[204]	[205]	[207]
Detection	Drunkenness	✓	✓	✓	✓	✓	✓																				✓
	Drowsiness							✓	✓	✓	✓	✓	✓												✓		✓
	Cellphone Hand-held													✓	✓	✓	✓	✓	✓	✓	✓	✓					✓
	Cellphone handsfree													✓	✓	✓				✓							
	Unsafe driving																						✓	✓	✓	✓	✓
Approach	Skin segmentation										✓										✓	✓					
	Viola-Jones							✓		✓		✓	✓								✓	✓	✓				
	Invariant features (SIFT, SURF, HOG, FV)										✓						✓	✓									
	Circular Hough Transform									✓	✓																
	Mathematical morphology											✓	✓														
	Nonlinear mapping between two time series (DTW)																							✓			
	Facial expression																		✓								
	Height and width of the mouth											✓											✓				
	Eyes' behavior							✓	✓		✓	✓	✓						✓				✓				
	Head orientation																		✓				✓				
	Movement detection																				✓	✓	✓	✓	✓	✓	✓
	Based on vehicle's position at lane																						✓		✓		✓
	Based on inertial sensors				✓											✓								✓	✓	✓	✓
	Pose and Skeleton, 3D															✓			✓								✓
	Arm/hand position																		✓			✓					✓
	Driving evaluation				✓	✓	✓																✓	✓	✓		✓
	Evaluation of biological signs	✓	✓	✓				✓																			
	Using vehicle telemetry data																									✓	✓
	Beeps of high frequency														✓												
	Block cell phone usage for a time													✓													
	Intrusive system							✓	✓											✓							
Idea	Driver monitoring	✓	✓	✓				✓	✓	✓	✓	✓	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓				✓
	Driving style				✓	✓	✓																✓	✓	✓	✓	✓
	Vehicle ignition interlock	✓																									
Constraint	Difficulty of operation in real world																				✓						
	Driver participates actively in the system				✓	✓		✓	✓					✓	✓	✓				✓				✓	✓		
	Need illumination (passive sensor)									✓	✓	✓	✓		✓			✓	✓		✓	✓	✓	✓	✓		
	Driver reeducation is not available																✓	✓									
	The driver may not be tested	✓												✓						✓							
	Driving a long-time perturbs the detection		✓																								
	Uses custom templates for each driver						✓																				
Sensors		1T	1P		1A 1G	#M		1C #E	8E	1C	1C	1C	1C		1!	1A 1G 1B 1!	1I	1C	1K	6E	1C	1C	2C	1C 1A 1G 1B 1@	1C 1! 1@ 1A	1@ #V	1K 1A 1G #V

Legend: ✓- Yes, C - Camera (RGB), I - Camera (IR), A - Accelerometer, G - Gyroscope, @ - GPS, T - Transdermal
K - Kinect, M - Mass sensor, B - Compass, P - Meter Pulse, ! - Microphone, E - Electrodes,
V - Vehicle telemetry, # - many.

light intensity captured by a passive device. The 3D Sensor can be installed in a position in front of the driver and that does not affect the vehicle's drivability, but thus providing the system with the ability to observe all the upper parts of the human body (head, neck, shoulders, arms and hands) in their usual positions when driving the vehicle, thus enabling the perception of unusual poses and actions for a driver in perfect driving conditions. Reference [206] is an example of a method that employs a 3D Sensor (Kinect v2) but uses sensor fusion and, therefore, is still dependent on RGB data and the incidence of the sun. Reference [207]'s work, on the other hand, uses only 3D information (based on a Kinect v2 sensor) to track the driver's pose and thus observe distractions and disturbances. By using only data from an active sensor (3D) it is resistant to solar interference and can operate during the night.

Cell phone use, in certain positions, may not be properly detected by the internal sensor used to monitor the driver. For example, when sending an SMS, the driver can leave the cell phone in a region outside the sensor's field of view, or even occluded by other objects. Telephone conversations using *hands-free* may not be detectable by the vehicle's internal data (driver movement). Despite the focus of this project's detection of cell phone use using the conventional way (holding the cell phone to the ear), it is possible to differentiate some situations of cognitive distraction caused by sending an SMS or *hands-free* conversation using the external characteristics that allow qualifying the driving of

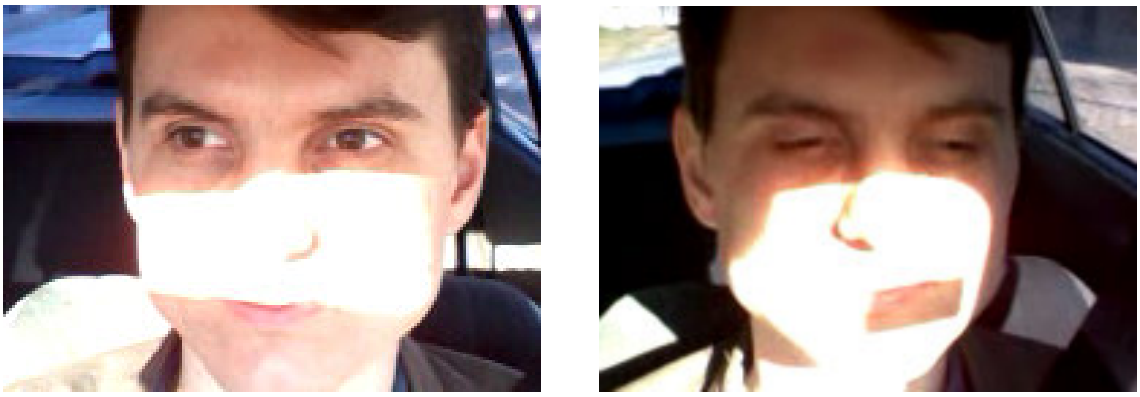


FIGURE 26. Examples of frames difficult to segment due to direct incidence of sunlight.

the vehicle and, therefore, noticing the change in the driving pattern.

The position of the data capture sensor in driver monitoring is extremely important, as it allows the correct functioning of the system and obtaining driver data within the most accurate region of the sensor. In addition, an inappropriate choice can make the system unfeasible in everyday situations. Reference [196]'s work uses a driver-side camera, positioned on the passenger door (inside the vehicle). Due to this choice, it is not possible to have a passenger sitting in front of the vehicle (between the camera and the driver), which results in a reduction in the vehicle's carrying capacity. For this reason, the position in front of the driver adopted in [197]'s work is the most common sensor position among the observed driver monitoring works.

The use of an intrusive system, as for example adopted by [185], [187], and [195], should also be avoided. Since this type of system can cause discomfort to the driver because they need electrodes or other equipment in contact with their skin, in addition to that, they obligatorily need the driver's consent for their operation.

The issue of driver consent is very important, as one should not need any assistance from the driver [176], [180], [185], [187], [193], [194], [195], [203], [204], i.e. the driver must not participate directly in the system to identify distractions and disturbances. The interesting thing is that the driver of the vehicle does not even notice that the system is in operation and observing his actions, thus tending to perform more natural and spontaneous actions.

Another relevant point in driver disturbances and distractions is re-education. An ADAS system should not just be limited to detecting improper driving by the driver, that is, the driver needs to be aware of his assessment in real time. Alerting the driver of his momentary difficulty driving, due to cell phone use or any other distraction, paves the way for his awareness that he needs to drive the vehicle with more responsibility. The works by [199] and [200] make it impossible to instantly re-educate the driver because they function as a traffic radar, that is, the driver would be alerted later when receiving traffic violation notices.

Another key point for a driver risk detection system is to ensure that the driver is being evaluated while driving. Systems such as [192], do not have any verification that a driver is being used. In the case of [195], the electrodes responsible for detecting distraction could be installed on a passenger, allowing risky driving situations to go unnoticed by the system. Reference [177]'s ignition lock system allows a passenger's blood alcohol concentration to be measured, that is, a passenger can impersonate a driver when the vehicle is started and after the vehicle is turned on system no longer performs tests.

The constitution of a generalist system, that is, capable of operating by evaluating data from drivers not used in training and in its constitution, is interesting. Systems such as [181] only operate with previously registered drivers, that is, all new drivers need to go through the creation of a personalized model that will enable the detection of risks while driving. In the [21] system, on the other hand, the driver's accuracy worsens when he spends too much time driving, that is, the pattern observed by the system changes, impairing the detection of driver disturbances and distractions.

Monitoring the driver and his way of driving the vehicle can benefit several businesses, such as bus companies and vehicle rental companies (insurers). Thus, you can have subsidies *on line* or *offline* (black box) of how employees or customers are managing vehicle fleets. Therefore, drivers who expose themselves to unnecessary risks can be penalized.

IV. CONCLUSION

In this article we present a literature review together with our work developed at the LRM (Mobile Robotics Laboratory) on detection, classification and tracking of obstacles in $2D$ and $3D$ images (external view) and also driver analysis systems (view inside the vehicle). Intelligent robotic vehicle applications require accuracy and real-time response, as they are defined as critical applications of embedded systems.

In the area of computer vision for vehicles, there are numerous researches from large development centers, but we still need to evolve further, enabling safe and flawless navigation.

Through the knowledge of our computer vision team, we believe that for the evolution of the state of the art in perception systems, we need to advance mainly with the models of algorithms applied for depth estimation, since we currently do not reach good accuracy in long distances (> 30 meters). We also need to advance in obstacle tracking models, making it possible to involve the temporal analysis of each situation evaluated in the scene. We also apply data collection and analysis in Brazilian traffic environments, making it possible to adapt our algorithms to particular situations that happen in our country in real data.

In ADAS systems, our proposal is to evaluate the driver and what happens outside the vehicle (external and internal perception together), making it possible to detect serious failures in the task of driving and supporting the avoidance of serious accidents in situations of drunkenness, drowsiness or human error.

REFERENCES

[1] J. M. Armingol, A. de la Escalera, C. Hilario, J. M. Collado, J. P. Carrasco, M. J. Flores, J. M. Pastor, and F. J. Rodríguez, "IVVI: Intelligent vehicle based on visual information," *Robot. Auto. Syst.*, vol. 55, no. 12, pp. 904–916, Dec. 2007.

[2] K. A. Brookhuis, D. De Waard, and W. H. Janssen, "Behavioural impacts of advanced driver assistance systems—An overview," *Eur. J. Transp. Infrastruct. Res.*, vol. 1, no. 3, pp. 245–253, 2019.

[3] A. Tapani, "Analysis of system effects of driver assistance systems by traffic simulation," in *Proc. Young Researchers Seminar Organised ECTRI, FEHRL FERSI*, 2007. [Online]. Available: http://www.ectri.org/YRS07/Papiers/Session-5/Tapani.pdf

[4] S. Lefévre, C. Laugier, and J. Ibañez-Guzmán, "Exploiting map information for driver intention estimation at road intersections," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2011, pp. 583–588.

[5] *SAE Levels of Driving Automation Refined for Clarityand International Audience*, SAE Int., Warrendale, PA, USA, 2021.

[6] N. Floudas, A. Amditis, A. Keinath, K. Bengler, and A. Engeln, "Review and taxonomy of IVIS/ADAS applications," Deliverable D2.1.2, Tech. Rep., Jan. 2005, vol. 1.

[7] P.-Y. Hsiao, C.-W. Yeh, S.-S. Huang, and L.-C. Fu, "A portable vision-based real-time lane departure warning system: Day and night," *IEEE Trans. Veh. Technol.*, vol. 58, no. 4, pp. 2089–2094, May 2009.

[8] E. Cardarelli, "Vision-based blind spot monitoring," in *Handbook of Intelligent Vehicles*, A. Eskandarian, Ed. London, U.K.: Springer, 2012, pp. 1071–1087.

[9] N. Manikoth, R. Loce, E. Bernal, and W. Wu, "Survey of computer vision in roadway transportation systems," *Proc. SPIE*, vol. 8305, Feb. 2012, Art. no. 83050W.

[10] P. F. Alcantarilla, L. M. Bergasa, P. Jiménez, I. Parra, D. F. Llorca, M. A. Sotelo, and S. S. Mayoral, "Automatic lightbeam controller for driver assistance," *Mach. Vis. Appl.*, vol. 22, no. 5, pp. 819–835, Sep. 2011.

[11] D. Marenco, D. Fontana, G. Ghisio, G. Monchiero, E. Cardarelli, P. Medici, and P. P. P. VisLab, "A validation tool for traffic signs recognition systems," in *Proc. 12th Int. IEEE Conf. Intell. Transp. Syst.*, Oct. 2009, pp. 1–6.

[12] I. Totzke, S. Jessberger, D. Muhlbacher, and H. P. Kruger, "Semi-autonomous advanced parking assistants: Do they really have to be learned if steering is automated?" *IET Intell. Transp. Syst.*, vol. 5, no. 2, pp. 141–147, Jun. 2011.

[13] M. Kutila, *Methods for Machine Vision Based Driver Monitoring Applications*, no. 621. Princeton, NJ, USA: Citeseer, 2006.

[14] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan, "The pothole patrol: Using a mobile sensor network for road surface monitoring," in *Proc. 6th Int. Conf. Mobile Syst., Appl., Services*, 2008, pp. 29–39.

[15] F. Tango, P. Carrea, and E. Gobetto, "The development of a smart pre-crash system-the methods for machine vision based driver monitoring applications project," in *Proc. 7th World Congr. Intell. Transp. Syst.*, 2000.

[16] K. Hojjati-Emami, B. S. Dhillon, and K. Jenab, "Reliability prediction for the vehicles equipped with advanced driver assistance systems (ADAS) and passive safety systems (PSS)," *Int. J. Ind. Eng. Comput.*, vol. 3, no. 5, pp. 731–742, Oct. 2012.

[17] D. Geronimo, A. Sappa, and A. López, "Stereo-based candidate generation for pedestrian protection systems," Tech. Rep., 2009.

[18] B. Kisacanin, "Automotive vision for advanced driver assistance systems," in *Proc. Int. Symp. VLSI Technol., Syst. Appl.*, Apr. 2011, pp. 1–2.

[19] R. Buckeridge, "With autonomous, self-driving cars likely to be commonplace by around 2025, these vehicles will change our roads, our relationship with our cars and society at large. Buckle up, a revolution is coming!" Tech. Rep.,2015.

[20] A. Amditis, M. Bimpas, G. Thomaidis, M. Tsogas, M. Netto, S. Mammar, A. Beutner, N. Mohler, T. Wirthgen, S. Zipser, A. Etemad, M. Da Lio, and R. Cicilloni, "A situation-adaptive lane-keeping support system: Overview of the SAFELANE approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 617–629, Sep. 2010.

[21] K. Murata, E. Fujita, S. Kojima, S. Maeda, Y. Ogura, T. Kamei, T. Tsuji, S. Kaneko, M. Yoshizumi, and N. Suzuki, "Noninvasive biological sensor system for detection of drunk driving," *IEEE Trans. Inf. Technol. Biomed.*, vol. 15, no. 1, pp. 19–25, Jan. 2011.

[22] L. C. Fernandes, J. R. Souza, G. Pessin, P. Y. Shinzato, D. Sales, C. Mendes, M. Prado, R. Klaser, A. C. Magalhães, A. Hata, D. Pigatto, K. C. Branco, V. Grassi, F. S. Osorio, and D. F. Wolf, "CaRINA intelligent robotic car: Architectural design and applications," *J. Syst. Archit.*, vol. 60, no. 4, pp. 372–392, Apr. 2014.

[23] A. Fisher. (2014). Inside Google's quest to popularize self-driving cars. Popular Science. Accessed: Jun. 11, 2015. [Online]. Available: http://www.popsci.com/cars/article/2013-09/google-self-driving-car

[24] E. Badger, "5 confounding questions that hold the key to the future of driverless cars," Tech. Rep., 2015.

[25] D. R. Bruno and F. S. Osorio, "Image classification system based on deep learning applied to the recognition of traffic signs for intelligent robotic vehicle navigation purposes," in *Proc. Latin Amer. Robot. Symp. (LARS) Brazilian Symp. Robot. (SBR)*, Nov. 2017, pp. 1–6.

[26] D. R. Bruno, D. O. Sales, J. Amaro, and F. S. Osório, "Analysis and fusion of 2D and 3D images applied for detection and recognition of traffic signs using a new method of features extraction in conjunction with deep learning," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2018, pp. 1–8.

[27] D. R. Bruno and F. S. Osorio, "A comparison of traffic signs detection methods in 2D and 3D images for the benefit of the navigation of autonomous vehicles," in *Proc. Latin Amer. Robot. Symp., Brazilian Symp. Robot. (SBR) Workshop Robot. Educ. (WRE)*, Nov. 2018, pp. 26–32.

[28] D. R. Bruno, T. C. Santos, J. A. R. Silva, D. F. Wolf, and F. S. Osório, "Advanced driver assistance system based on automated routines for the benefit of human faults correction in robotics vehicles," in *Proc. Latin Amer. Robot. Symp., Brazilian Symp. Robot. (SBR) Workshop Robot. Educ. (WRE)*, Nov. 2018, pp. 112–117.

[29] D. R. Bruno, I. P. Gomes, F. S. Osório, and D. F. Wolf, "Advanced driver assistance system based on NeuroFSM applied in the detection of autonomous human faults and support to semi-autonomous control for robotic vehicles," in *Proc. Latin Amer. Robot. Symp. (LARS), Brazilian Symp. Robot. (SBR) Workshop Robot. Educ. (WRE)*, Oct. 2019, pp. 92–97.

[30] D. R. Bruno and F. S. Osório, "Visual attention system based on fuzzy classifier to define priority of traffic signs for intelligent robotic vehicle navigation purposes," in *Proc. 19th Int. Conf. Adv. Robot. (ICAR)*, Dec. 2019, pp. 434–440.

[31] D. Bruno, L. P. N. Matias, J. Amaro, F. S. Osório, and D. Wolf, "Computer vision system with 2D and 3D data fusion for detection of possible auxiliaries routes in stretches of interdicted roads," in *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, 2019, pp. 7372–7381.

[32] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[33] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[34] R. Gavrilescu, C. Zet, C. Fosălau, M. Skoczylas, and D. Cotovanu, "Faster R-CNN: An approach to real-time object detection," in *Proc. Int. Conf. Expo. Electr. Power Eng. (EPE)*, Oct. 2018, pp. 165–168.

[35] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[36] S. Chen, J. Hong, T. Zhang, J. Li, and Y. Guan, "Object detection using deep learning: Single shot detector with a refined feature-fusion structure," in *Proc. IEEE Int. Conf. Real-Time Comput. Robot. (RCAR)*, Aug. 2019, pp. 219–224.

[37] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6517–6525.

[38] S. Srivastava, F. Jurie, and G. Sharma, "Learning 2D to 3D lifting for object detection in 3D for autonomous vehicles," 2019, *arXiv:1904.08494*.

[39] S. McCrae and A. Zakhor, "3D object detection for autonomous driving using temporal LiDAR data," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Oct. 2020, pp. 2661–2665.

[40] I. Baek, T.-C. Tai, M. M. Bhat, K. Ellango, T. Shah, K. Fuseini, and R. R. Rajkumar, "CurbScan: Curb detection and tracking using multi-sensor fusion," in *Proc. IEEE 23rd Int. Conf. Intell. Transp. Syst. (ITSC)*, Sep. 2020, pp. 1–8.

[41] I.-S. Weon, S.-G. Lee, and J.-K. Ryu, "Object recognition based interpolation with 3D LiDAR and vision for autonomous driving of an intelligent vehicle," *IEEE Access*, vol. 8, pp. 65599–65608, 2020.

[42] V. Guizilini, J. Li, R. Ambrus, and A. Gaidon, "Geometric unsupervised domain adaptation for semantic segmentation," 2021, *arXiv:2103.16694v2*.

[43] D. O. Sales, J. Amaro, and F. S. Osório, "3D shape descriptor for objects recognition," in *Proc. Latin Amer. Robot. Symp. (LARS) Brazilian Symp. Robot. (SBR)*, Nov. 2017, pp. 1–6.

[44] J. Chen, P. Zhao, H. Liang, and T. Mei, "A multiple attribute-based decision making model for autonomous vehicle in urban environment," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2014, pp. 480–485.

[45] I. P. Gomes, D. R. Bruno, F. S. Osório, and D. F. Wolf, "Diagnostic analysis for an autonomous truck using multiple attribute decision making," in *Proc. Latin Amer. Robot. Symp., Brazilian Symp. Robot. (SBR) Workshop Robot. Educ. (WRE)*, Nov. 2018, pp. 283–290.

[46] M. Seyedhosseini and T. Tasdizen, "Semantic image segmentation with contextual hierarchical models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 5, pp. 951–964, May 2016.

[47] N. Silberman and R. Fergus, "Indoor scene segmentation using a structured light sensor," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCV Workshops)*, Nov. 2011, pp. 601–608.

[48] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[49] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention—MICCAI 2015*, N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi, Eds. Cham, Switzerland: Springer, 2015, pp. 234–241.

[50] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder–decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[51] S. Hao, Y. Zhou, Y. Guo, R. Hong, J. Cheng, and M. Wang, "Real-time semantic segmentation via spatial-detail guided context propagation," *IEEE Trans. Neural Netw. Learn. Syst.*, Mar. 2022, pp. 1–12.

[52] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," *CoRR*, vol. abs/1511.07122, pp. 1–13, Nov. 2016.

[53] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 764–773.

[54] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 6230–6239.

[55] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7794–7803.

[56] L. Chen, Z. Lin, Z. Wang, Y. Yang, and M. Cheng, "Spatial information guided convolution for real-time RGBD semantic segmentation," *IEEE Trans. Image Process.*, vol. 30, pp. 2313–2324, 2021.

[57] W. Shi, J. Xu, D. Zhu, G. Zhang, X. Wang, J. Li, and X. Zhang, "RGB-D semantic segmentation and label-oriented voxelgrid fusion for accurate 3D semantic mapping," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 1, pp. 183–197, Jan. 2022.

[58] Y. Li, L. Song, Y. Chen, Z. Li, X. Zhang, X. Wang, and J. Sun, "Learning dynamic routing for semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8550–8559.

[59] H. Li, P. Xiong, H. Fan, and J. Sun, "DFANet: Deep feature aggregation for real-time semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9514–9523.

[60] Y. Zhu, K. Sapra, F. A. Reda, K. J. Shih, S. Newsam, A. Tao, and B. Catanzaro, "Improving semantic segmentation via video propagation and label relaxation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8848–8857.

[61] Y. Yuan, L. Wang, and Y. Wang, "CSANet for video semantic segmentation with inter-frame mutual learning," *IEEE Signal Process. Lett.*, vol. 28, pp. 1675–1679, 2021.

[62] A. Chaurasia and E. Culurciello, "LinkNet: Exploiting encoder representations for efficient semantic segmentation," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, 2017, pp. 1–4.

[63] M. Oršić and S. Šegvić, "Efficient semantic segmentation with pyramidal fusion," *Pattern Recognit.*, vol. 110, Feb. 2021, Art. no. 107611.

[64] M. Kim, B. Park, and S. Chi, "Accelerator-aware fast spatial feature network for real-time semantic segmentation," *IEEE Access*, vol. 8, pp. 226524–226537, 2020.

[65] T. Emara, H. E. A. E. Munim, and H. M. Abbas, "LiteSeg: A novel lightweight convNet for semantic segmentation," in *Proc. Digit. Image Comput., Techn. Appl. (DICTA)*, 2019, pp. 1–7.

[66] H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia, "ICNet for real-time semantic segmentation on high-resolution images," in *Computer Vision—ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham, Switzerland: Springer, 2018, pp. 418–434.

[67] C. Yu, C. Gao, J. Wang, G. Yu, C. Shen, and N. Sang, "BiSeNet V2: Bilateral network with guided aggregation for real-time semantic segmentation," *Int. J. Comput. Vis.*, vol. 129, no. 11, pp. 3051–3068, Nov. 2021.

[68] C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang, "BiSeNet: Bilateral segmentation network for real-time semantic segmentation," in *Computer Vision—ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham, Switzerland: Springer, 2018, pp. 334–349.

[69] H. Si, Z. Zhang, F. Lv, G. Yu, and F. Lu, "Real-time semantic segmentation via multiply spatial fusion network," 2019, *arXiv:1911.07217*.

[70] P. Hu, F. Perazzi, F. C. Heilbron, O. Wang, Z. Lin, K. Saenko, and S. Sclaroff, "Real-time semantic segmentation with fast attention," *IEEE Robot. Autom. Lett.*, vol. 6, no. 1, pp. 263–270, Jan. 2021.

[71] J. Sun, S. Jung, and S. Ko, "Lightweight prediction and boundary attention-based semantic segmentation for road scene understanding," *IEEE Access*, vol. 8, pp. 108449–108460, 2020.

[72] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[73] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L. Chen, "MobileNetV2: Inverted residuals and linear bottlenecks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4510–4520.

[74] A. Howard, M. Sandler, B. Chen, W. Wang, L. Chen, M. Tan, G. Chu, V. Vasudevan, Y. Zhu, R. Pang, H. Adam, and Q. Le, "Searching for MobileNetV3," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1314–1324.

[75] Q. Xu, Y. Ma, J. Wu, and C. Long, "Faster BiSeNet: A faster bilateral segmentation network for real-time semantic segmentation," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2021, pp. 1–8.

[76] M. Fan, S. Lai, J. Huang, X. Wei, Z. Chai, J. Luo, and X. Wei, "Rethinking BiSeNet for real-time semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 9711–9720.

[77] Y. Li, X. Li, C. Xiao, H. Li, and W. Zhang, "EACNet: Enhanced asymmetric convolution for real-time semantic segmentation," *IEEE Signal Process. Lett.*, vol. 28, pp. 234–238, 2021.

[78] R. P. K. Poudel, S. Liwicki, and R. Cipolla, "Fast-SCNN: Fast semantic segmentation network," in *Proc. BMVC*, 2019, pp. 1–12.
[79] M. Paul, C. Mayer, L. Van Gool, and R. Timofte, "Efficient video semantic segmentation with labels propagation and refinement," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2020, pp. 2862–2871.
[80] X. Chen, A. Wu, and Y. Han, "Capturing the spatio-temporal continuity for video semantic segmentation," *IET Image Process.*, vol. 13, no. 14, pp. 2813–2820, Dec. 2019.
[81] Y. Li, J. Shi, and D. Lin, "Low-latency video semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 5997–6005.
[82] Q. Lv, X. Sun, C. Chen, J. Dong, and H. Zhou, "Parallel complement network for real-time semantic segmentation of road scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 5, pp. 4432–4444, May 2022.
[83] H. Wang, W. Wang, and J. Liu, "Temporal memory attention for video semantic segmentation," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2021, pp. 2254–2258.
[84] J. Zhuang, J. Yang, L. Gu, and N. Dvornek, "ShelfNet for fast semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Oct. 2019, pp. 847–856.
[85] D. Oh, D. Ji, C. Jang, Y. Hyun, H. S. Bae, and S. Hwang, "Segmenting 2K-videos at 36.5 FPS with 24.3 GFLOPs: Accurate and lightweight realtime semantic segmentation network," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 3153–3160.
[86] W. Wang, "Semi-supervised semantic segmentation network based on knowledge distillation," in *Proc. IEEE 4th Adv. Inf. Manage., Communicates, Electron. Autom. Control Conf. (IMCEC)*, vol. 4, Jun. 2021, pp. 1900–1905.
[87] P. Hu, F. Caba, O. Wang, Z. Lin, S. Sclaroff, and F. Perazzi, "Temporally distributed networks for fast video semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8815–8824.
[88] H. Lu and Z. Deng, "A boundary-aware distillation network for compressed video semantic segmentation," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 5354–5359.
[89] Y. Hong, H. Pan, W. Sun, and Y. Jia, "Deep dual-resolution networks for real-time and accurate semantic segmentation of road scenes," 2021, *arXiv:2101.06085*.
[90] M. Liu and H. Yin, "Efficient pyramid context encoding and feature embedding for semantic segmentation," *Image Vis. Comput.*, vol. 111, Jul. 2021, Art. no. 104195.
[91] W. Chen, X. Gong, X. Liu, Q. Zhang, Y. Li, and Z. Wang, "FasterSeg: Searching for faster real-time semantic segmentation," 2019, *arXiv:1912.10917*.
[92] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1800–1807.
[93] E. Romera, J. M. Álvarez, L. M. Bergasa, and R. Arroyo, "ERFNet: Efficient residual factorized ConvNet for real-time semantic segmentation," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 263–272, Jan. 2018.
[94] A. Luo, F. Yang, X. Li, R. Huang, and H. Cheng, "EKENet: Efficient knowledge enhanced network for real-time scene parsing," *Pattern Recognit.*, vol. 111, Mar. 2021, Art. no. 107671.
[95] L. Rosas-Arias, G. Benitez-Garcia, J. Portillo-Portillo, G. Sánchez-Pérez, and K. Yanai, "Fast and accurate real-time semantic segmentation with dilated asymmetric convolutions," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 2264–2271.
[96] S. Mehta, M. Rastegari, A. Caspi, L. Shapiro, and H. Hajishirzi, "ESPNet: Efficient spatial pyramid of dilated convolutions for semantic segmentation," in *Computer Vision—ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham, Switzerland: Springer, 2018, pp. 561–580.
[97] Z. Zhu, M. Xu, S. Bai, T. Huang, and X. Bai, "Asymmetric non-local neural networks for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 593–602.
[98] M. Paul, M. Danelljan, L. V. Gool, and R. Timofte, "Local memory attention for fast video semantic segmentation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2021, pp. 1102–1109.
[99] J. Li, W. Wang, J. Chen, L. Niu, J. Si, C. Qian, and L. Zhang, "Video semantic segmentation via sparse temporal transformer," in *Proc. 29th ACM Int. Conf. Multimedia*. New York, NY, USA, Association for Computing Machinery, Oct. 2021, pp. 59–68.
[100] W. Wang and U. Neumann, "Depth-aware CNN for RGB-D segmentation," in *Computer Vision—ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham, Switzerland: Springer, 2018, pp. 144–161.
[101] A. Cardace, L. De Luigi, P. Z. Ramirez, S. Salti, and L. Di Stefano, "Plugging self-supervised monocular depth into unsupervised domain adaptation for semantic segmentation," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2022, pp. 1999–2009.
[102] *Assume Self-Driving Cars are a Hacker's Dream? Think Again*, 2017.
[103] L. Li, B. Qian, J. Lian, W. Zheng, and Y. Zhou, "Traffic scene segmentation based on RGB-D image and deep learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 5, pp. 1664–1669, May 2018.
[104] L. Li, W. Zheng, L. Kong, U. Ozguner, W. Hou, and J. Lian, "Real-time traffic scene segmentation based on multi-feature map and deep learning," in *Proc. IEEE Intell. Vehicle Symp. (IV)*, Jun. 2018, pp. 7–12.
[105] C. Hazirbas, L. Ma, C. Domokos, and D. Cremers, "Fusenet: Incorporating depth into semantic segmentation via fusion-based CNN architecture," in *Computer Vision—ACCV 2016*, S.-H. Lai, V. Lepetit, K. Nishino, and Y. Sato, Eds. Cham, Switzerland: Springer, 2017, pp. 213–228.
[106] Y. Qian, L. Deng, T. Li, C. Wang, and M. Yang, "Gated-residual block for semantic segmentation using RGB-D data," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 11836–11844, Aug. 2022.
[107] X. Zhang, Y. Chen, H. Zhang, S. Wang, J. Lu, and J. Yang, "When visual disparity generation meets semantic segmentation: A mutual encouragement approach," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 3, pp. 1853–1867, Mar. 2021.
[108] J. Zhang, K. A. Skinner, R. Vasudevan, and M. Johnson-Roberson, "DispSegNet: Leveraging semantics for end-to-end learning of disparity estimation from stereo imagery," *IEEE Robot. Autom. Lett.*, vol. 4, no. 2, pp. 1162–1169, Apr. 2019.
[109] L. Zhou, H. Zhang, Y. Long, L. Shao, and J. Yang, "Depth embedded recurrent predictive parsing network for video scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 12, pp. 4643–4654, Dec. 2019.
[110] M. D. Ansari, S. Krauß, O. Wasenmüller, and D. Stricker, "ScaleNet: Scale invariant network for semantic segmentation in urban driving scenes," in *Proc. 13th Int. Joint Conf. Comput. Vis., Imag. Comput. Graph. Theory Appl.*, 2018, pp. 399–404.
[111] S. Kong and C. Fowlkes, "Recurrent scene parsing with perspective understanding in the loop," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 956–965.
[112] L. Hoyer, D. Dai, Y. Chen, A. Köring, S. Saha, and L. Van Gool, "Three ways to improve semantic segmentation with self-supervised depth estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 11125–11135.
[113] Q. Wang, D. Dai, L. Hoyer, L. Van Gool, and O. Fink, "Domain adaptive semantic segmentation with self-supervised depth estimation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 8495–8505.
[114] M. Aladem and S. A. Rawashdeh, "A single-stream segmentation and depth prediction CNN for autonomous driving," *IEEE Intell. Syst.*, vol. 36, no. 4, pp. 79–85, Jul. 2021.
[115] K. Goel, P. Srinivasan, S. Tariq, and J. Philbin, "QuadroNet: Multi-task learning for real-time semantic depth aware instance segmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 315–324.
[116] D. Xu, W. Ouyang, X. Wang, and N. Sebe, "PAD-Net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 675–684.
[117] H. Jiang, G. Larsson, M. Maire, G. Shakhnarovich, and E. Learned-Miller, "Self-supervised relative depth learning for urban scene understanding," in *Computer Vision—ECCV 2018*, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds. Cham, Switzerland: Springer, 2018, pp. 20–37.
[118] P. Z. Ramirez, A. Tonioni, S. Salti, and L. D. Stefano, "Learning across tasks and domains," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 8109–8118.
[119] R. Chavhan, A. Jha, B. Banerjee, and S. Chaudhuri, "ADA-AT/DT: An adversarial approach for cross-domain and cross-task knowledge transfer," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2021, pp. 3501–3510.
[120] S. Papadopoulos, I. Mademlis, and I. Pitas, "Semantic image segmentation guided by scene geometry," in *Proc. IEEE Int. Conf. Auto. Syst. (ICAS)*, Aug. 2021, pp. 1–5.

[121] A. Loukkal, Y. Grandvalet, and Y. Li, "Disparity weighted loss for semantic segmentation of driving scenes," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 3427–3432.

[122] E. Shelhamer, K. Rakelly, J. Hoffman, and T. Darrell, "Clockwork ConvNets for video semantic segmentation," in *Computer Vision—ECCV 2016*, G. Hua and H. Jégou, Eds. Cham, Switzerland: Springer, 2018, pp. 852–868.

[123] S. Jain, X. Wang, and J. E. Gonzalez, "Accel: A corrective fusion network for efficient semantic segmentation on video," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8858–8867.

[124] Y. Xu, T. Fu, H. Yang, and C. Lee, "Dynamic video segmentation network," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6556–6565.

[125] T. Zhou, F. Porikli, D. Crandall, L. Van Gool, and W. Wang, "A survey on deep learning technique for video segmentation," 2021, *arXiv:2107.01153*.

[126] Y. Liu, C. Shen, C. Yu, and J. Wang, "Efficient semantic video segmentation with per-frame inference," in *Computer Vision—ECCV 2020*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Cham, Switzerland: Springer, 2020, pp. 352–368.

[127] J. Xiong, L. Po, W. Yu, Y. Zhao, and K. Cheung, "Distortion map-guided feature rectification for efficient video semantic segmentation," *IEEE Trans. Multimedia*, vol. 25, pp. 1019–1032, 2023.

[128] J. Zhuang, Z. Wang, and B. Wang, "Video semantic segmentation with distortion-aware feature correction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 8, pp. 3128–3139, Aug. 2021.

[129] J. Wu, Z. Wen, S. Zhao, and K. Huang, "Video semantic segmentation via feature propagation with holistic attention," *Pattern Recognit.*, vol. 104, Aug. 2020, Art. no. 107268.

[130] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[131] S. Varghese, S. Gujamagadi, M. Klingner, N. Kapoor, A. Bär, J. D. Schneider, K. Maag, P. Schlicht, F. Hüger, and T. Fingscheidt, "An unsupervised temporal consistency (TC) loss to improve the performance of semantic segmentation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 12–20.

[132] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Computer Vision—ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham, Switzerland: Springer, 2014, pp. 740–755.

[133] S. Hwang, J. Park, N. Kim, Y. Choi, and I. S. Kweon, "Multispectral pedestrian detection: Benchmark dataset and baseline," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1037–1045.

[134] A. Rasouli, I. Kotseruba, and J. K. Tsotsos, "Are they going to cross? A benchmark dataset and baseline for pedestrian crosswalk behavior," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2017, pp. 206–213.

[135] X. Li, F. Flohr, Y. Yang, H. Xiong, M. Braun, S. Pan, K. Li, and D. M. Gavrila, "A new benchmark for vision-based cyclist detection," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2016, pp. 1028–1033.

[136] S. R. Richter, V. Vineet, S. Roth, and V. Koltun, "Playing for data: Ground truth from computer games," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2016, pp. 102–118.

[137] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. Lopez, "The SYNTHIA Dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 3234–3243.

[138] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.

[139] W. Maddern, G. Pascoe, C. Linegar, and P. Newman, "1 year, 1000 km: The Oxford robotcar dataset," *Int. J. Robot. Res.*, vol. 36, no. 1, pp. 3–15, 2017.

[140] K. Takumi, K. Watanabe, Q. Ha, A. Tejero-De-Pablos, Y. Ushiku, and T. Harada, "Multispectral object detection for autonomous vehicles," in *Proc. Thematic Workshops ACM Multimedia*. New York, NY, USA, Association for Computing Machinery, Oct. 2017, pp. 35–43.

[141] Q. Ha, K. Watanabe, T. Karasawa, Y. Ushiku, and T. Harada, "MFNet: Towards real-time semantic segmentation for autonomous vehicles with multi-spectral scenes," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2017, pp. 5108–5115.

[142] G. Neuhold, T. Ollmann, S. R. Bulò, and P. Kontschieder, "The mapillary vistas dataset for semantic understanding of street scenes," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 5000–5009.

[143] Y. Choi, N. Kim, S. Hwang, K. Park, J. S. Yoon, K. An, and I. S. Kweon, "KAIST multi-spectral day/night data set for autonomous and assisted driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 3, pp. 934–948, Mar. 2018.

[144] X. Huang, P. Wang, X. Cheng, D. Zhou, Q. Geng, and R. Yang, "The ApolloScape open dataset for autonomous driving and its application," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 10, pp. 2702–2719, Oct. 2020.

[145] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. Erin Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuScenes: A multimodal dataset for autonomous driving," 2019, *arXiv:1903.11027*.

[146] M. Bijelic, T. Gruber, F. Mannan, F. Kraus, W. Ritter, K. Dietmayer, and F. Heide, "Seeing through fog without seeing fog: Deep multimodal sensor fusion in unseen adverse weather," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 11679–11689.

[147] J. Xue, J. Fang, T. Li, B. Zhang, P. Zhang, Z. Ye, and J. Dou, "BLVD: Building a large-scale 5D semantics benchmark for autonomous driving," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 6685–6691.

[148] P. Sun, H. Kretzschmar, X. Dotiwalla, A. Chouard, V. Patnaik, P. Tsui, J. Guo, Y. Zhou, Y. Chai, B. Caine, V. Vasudevan, W. Han, J. Ngiam, H. Zhao, A. Timofeev, S. Ettinger, M. Krivokon, A. Gao, A. Joshi, Y. Zhang, J. Shlens, Z. Chen, and D. Anguelov, "Scalability in perception for autonomous driving: Waymo open dataset," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2443–2451.

[149] A. Patil, S. Malla, H. Gang, and Y. Chen, "The H3D dataset for full-surround 3D multi-object detection and tracking in crowded urban scenes," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 9552–9557.

[150] J. Geyer, Y. Kassahun, M. Mahmudi, X. Ricou, R. Durgesh, A. S. Chung, L. Hauswald, V. Hoang Pham, M. Mühlegg, S. Dorn, T. Fernandez, M. Jänicke, S. Mirashi, C. Savani, M. Sturm, O. Vorobiov, M. Oelker, S. Garreis, and P. Schuberth, "A2D2: Audi autonomous driving dataset," 2020, *arXiv:2004.06320*.

[151] Q. Pham, P. Sevestre, R. S. Pahwa, H. Zhan, C. H. Pang, Y. Chen, A. Mustafa, V. Chandrasekhar, and J. Lin, "A 3D dataset: Towards autonomous driving in challenging environments," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 2267–2273.

[152] M. Braun, S. Krebs, F. Flohr, and D. M. Gavrila, "EuroCity persons: A novel benchmark for person detection in traffic scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1844–1861, Aug. 2019.

[153] M. Chang, J. Lambert, P. Sangkloy, J. Singh, S. Bak, A. Hartnett, D. Wang, P. Carr, S. Lucey, D. Ramanan, and J. Hays, "Argoverse: 3D tracking and forecasting with rich maps," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8740–8749.

[154] R. Chan, K. Lis, S. Uhlemeyer, H. Blum, S. Honari, R. Siegwart, P. Fua, M. Salzmann, and M. Rottmann, "SegmentMeIfYouCan: A benchmark for anomaly segmentation," 2021, *arXiv:2104.14812*.

[155] A. Ligocki, A. Jelinek, and L. Zalud, "Brno urban dataset-the new data for self-driving agents and mapping tasks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 3284–3290.

[156] M. Pitropov, D. E. Garcia, J. Rebello, M. Smart, C. Wang, K. Czarnecki, and S. Waslander, "Canadian adverse driving conditions dataset," *Int. J. Robot. Res.*, vol. 40, nos. 4–5, pp. 681–690, Apr. 2021.

[157] F. Yu, H. Chen, X. Wang, W. Xian, Y. Chen, F. Liu, V. Madhavan, and T. Darrell, "BDD100K: A diverse driving dataset for heterogeneous multitask learning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2633–2642.

[158] T. Lim, S. A. Markowitz, and M. N. Do, "RaDICaL: A synchronized FMCW radar, depth, IMU and RGB camera data dataset with low-level FMCW radar signals," *IEEE J. Sel. Topics Signal Process.*, vol. 15, no. 4, pp. 941–953, Jun. 2021.

[159] P. Y. Shinzato and D. F. Wolf, "Statistical analysis of image-features used as inputs of an road identifier based in artificial neural networks," in *Proc. Latin Amer. Robot. Symp. Intell. Robot. Meeting*, Oct. 2010, pp. 19–24.

[160] P. Y. Shinzato and D. F. Wolf, "A road following approach using artificial neural networks combinations," *J. Intell., Robot. Syst.*, vol. 62, nos. 3–4, pp. 527–546, Jun. 2011.

[161] G. B. Eboli, P. Y. Shinzato, and D. F. Wolf, "Análise de atributos de imagem em segmentação baseada em blocos utilizando redes neurais artificiais," in *Proc. Anais Congresso Brasileiro Inteligência Computacional*, G. D. A. Barreto and J. A. F. Costa, Eds., Fortaleza, Brazil, Mar. 2016, pp. 1–8.

[162] P. Y. Shinzato, V. Grassi, F. S. Osorio, and D. F. Wolf, "Fast visual road recognition and horizon detection using multiple artificial neural networks," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2012, pp. 1090–1095.

[163] M. P. Batista, P. Y. Shinzato, D. F. Wolf, and D. Gomes, "Lane detection and estimation using perspective image," in *Proc. Joint Conf. Robot., SBR-LARS Robot. Symp. Robocontrol*, Oct. 2014, pp. 25–30.

[164] M. A. Dias and F. S. Osorio, "Fixed-point neural network ensembles for visual navigation," in *Proc. Brazilian Robot. Symp. Latin Amer. Robot. Symp.*, Oct. 2012, pp. 307–312.

[165] J. R. Souza, G. Pessin, P. Y. Shinzato, F. S. Osòrio, and D. F. Wolf, "Vision-based autonomous navigation using neural networks and templates in urban environments," in *Proc. 1st Brazilian Conf. Crit. Embedded Syst.*, 2011, pp. 55–60.

[166] D. O. Sales, L. C. Fernandes, F. S. Osòrio, and D. F. Wolf, "FSM-based visual navigation for autonomous vehicles," in *Proc. Workshop Vis. Control Mobile Robots-IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Vilamoura, Portugal, 2012.

[167] D. O. Sales and F. S. Osório, "Vision-based autonomous topological navigation in outdoor environments," in *Proc. Brazilian Robot. Symp. Latin Amer. Robot. Symp.*, Oct. 2012, pp. 91–96.

[168] J. R. Souza, D. O. Sales, P. Y. Shinzato, F. S. Osorio, and D. F. Wolf, "Template-based autonomous navigation and obstacle avoidance in urban environments," *ACM SIGAPP Appl. Comput. Rev.*, vol. 11, no. 4, pp. 49–59, Dec. 2011.

[169] P. Y. Shinzato, D. Gomes, and D. F. Wolf, "Road estimation with sparse 3D points from stereo data," in *Proc. 17th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2014, pp. 1688–1693.

[170] P. Y. Shinzato, C. Mendes, F. Osorio, and D. F. Wolf, "Performance evaluation of different machine learning techniques with stereo vision used to road detection task," in *Proc. Anais Congresso Brasileiro de Inteligência Computacional*, G. D. A. Barreto and J. A. F. Costa, Eds., Fortaleza, Brazil, Mar. 2016, pp. 1–8.

[171] P. Y. Shinzato, D. F. Wolf, and C. Stiller, "Road terrain detection: Avoiding common obstacle detection assumptions using sensor fusion," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2014, pp. 687–692.

[172] C. C. T. Mendes, V. Frémont, and D. F. Wolf, "Vision-based road detection using contextual blocks," in *Proc. 7th Workshop Planning, Perception Navigat. Intell. Vehicles (IEEE IROS)*, Hamburg, Germany, Sep. 2015, pp. 1–6.

[173] C. C. T. Mendes, V. Frémont, and D. F. Wolf, "Exploiting fully convolutional neural networks for fast road detection," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2016, pp. 3174–3179.

[174] L. R. T. Horita, A. T. M. Nakamura, D. F. Wolf, and V. G. Junior, "Improving multi-goal and target-driven reinforcement learning with supervised auxiliary task," in *Proc. 20th Int. Conf. Adv. Robot. (ICAR)*, Dec. 2021, pp. 290–295.

[175] E. Haile, "Drunk driver detection system," U.S. Patent 4 716 413, Dec. 29, 1987.

[176] E. Haile, "Drunk driver detection system," U.S. Patent 5 096 329, Mar. 17, 1992.

[177] J. Carroll, D. Bellehumeur, and C. Carroll, "System and method for detecting and measuring ethyl alcohol in the blood of a motorized vehicle driver transdermally and non-invasively in the presence of interferents," U.S. Patent App. 20 130 027 209, Jan. 31, 2013.

[178] D. S. Edmonds and J. W. Hopta, "Driver alcohol ignition interlock," U.S. Patent 6 229 908, May 8, 2001.

[179] P. N. Kathar and D. L. Bhuyar, "Design and implementation of driver drowsiness and alcohol intoxication detection using Raspberry PI," *Int. J. Innov. Res. Comput. Commun. Eng.*, vol. 4, pp. 14617–14625, Aug. 2016.

[180] J. Dai, J. Teng, X. Bai, Z. Shen, and D. Xuan, "Mobile phone based drunk driving detection," in *Proc. 4th Int. ICST Conf. Pervasive Comput. Technol. Healthcare*, 2010, pp. 1–8.

[181] M. M. Shirazi and A. B. Rad, "Detection of intoxicated drivers using online system identification of steering behavior," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 4, pp. 1738–1747, Aug. 2014.

[182] B. Akrout and W. Mahdi, "A visual based approach for drowsiness detection," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2013, pp. 1324–1329.

[183] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comput. Vis.*, vol. 57, no. 2, pp. 137–154, May 2004.

[184] J. Cauchie, V. Fiolet, and D. Villers, "Optimization of an Hough transform algorithm for the search of a center," *Pattern Recognit.*, vol. 41, no. 2, pp. 567–574, Feb. 2008.

[185] P. Ebrahim, A. Abdellaoui, W. Stolzmann, and B. Yang, "Eyelid-based driver state classification under simulated and real driving conditions," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2014, pp. 3190–3196.

[186] R. F. Knipling and W. W. Wierwille, "Vehicle-based drowsy driver detection: Current status and future prospects," in *Proc. IVHS Amer. Conf. Moving Toward Deployment*, 1994, pp. 1–23.

[187] M. Ben Dkhil, M. Neji, A. Wali, and A. M. Alimi, "A new approach for a safe car assistance system," in *Proc. 4th Int. Conf. Adv. Logistics Transp. (ICALT)*, May 2015, pp. 217–222.

[188] A. A. Lenskiy and J.-S. Lee, "Driver's eye blinking detection using novel color and texture segmentation algorithms," *Int. J. Control, Autom. Syst.*, vol. 10, no. 2, pp. 317–327, Apr. 2012.

[189] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vis. Image Understand.*, vol. 110, no. 3, pp. 346–359, Jun. 2008.

[190] K. Kumar, M. Alkoffash, S. Dange, A. Idarrou, N. Sridevi, J. Sheeba, N. Shah, S. Sharma, G. Elyasi, and H. Saremi, "Morphology based facial feature extraction and facial expression recognition for driver vigilance," *Int. J. Comput. Appl.*, vol. 51, no. 2, pp. 17–24, Aug. 2012.

[191] R. A. Berri, A. G. Silva, R. Arthur, and E. Girardi, "Detecção automática de sonolência em condutores de veículos utilizando imagens amplas e de baixa resolução," *Comput. Beach*, pp. 21–30, 2013.

[192] B. M. D. Cidades, "Aplicativo mãos NO volante," Tech. Rep., 2012.

[193] J. Yang, S. Sidhom, G. Chandrasekaran, T. Vu, H. Liu, N. Cecan, Y. Chen, M. Gruteser, and R. P. Martin, "Detecting driver phone use leveraging car speakers," in *Proc. 17th Annu. Int. Conf. Mobile Comput. Netw.*, 2011, pp. 97–108.

[194] H. Park, D. Ahn, T. Park, and K. G. Shin, "Automatic identification of driver's smartphone exploiting common vehicle-riding actions," *IEEE Trans. Mobile Comput.*, vol. 17, no. 2, pp. 265–278, Feb. 2018.

[195] S. V. Deshmukh and O. Dehzangi, "ECG-based driver distraction identification using wavelet packet transform and discriminative kernel-based features," in *Proc. IEEE Int. Conf. Smart Comput. (SMARTCOMP)*, May 2017, pp. 1–7.

[196] H. Veeraraghavan, N. Bird, S. Atev, and N. Papanikolopoulos, "Classifiers for driver activity monitoring," *Transp. Res. C, Emerg. Technol.*, vol. 15, no. 1, pp. 51–67, Feb. 2007.

[197] R. Berri, F. Osorio, R. Parpinelli, and A. Silva, "A hybrid vision system for detecting use of mobile phones while driving," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2016, pp. 4601–4610.

[198] M.-K. Hu, "Visual pattern recognition by moment invariants," *IEEE Trans. Inf. Theory*, vol. IT-8, no. 2, pp. 179–187, Feb. 1962.

[199] Y. Artan, O. Bulan, R. P. Loce, and P. Paul, "Driver cell phone usage detection from HOV/HOT NIR images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2014, pp. 225–230.

[200] K. Seshadri, F. Juefei-Xu, D. K. Pal, M. Savvides, and C. P. Thor, "Driver cell phone usage detection on strategic highway research program (SHRP2) face view videos," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2015, pp. 35–43.

[201] J. Lee, J. Li, L. Liu, and C. Chen, "A novel driving pattern recognition and status monitoring system," in *Advances in Image and Video Technology* (Lecture Notes in Computer Science), vol. 4319, L. Chang and W. Lie, Eds. Berlin, Germany: Springer, 2006, pp. 504–512.

[202] D. Mitrovic, "Reliable method for driving events recognition," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 2, pp. 198–205, Jun. 2005.

[203] D. A. Johnson and M. M. Trivedi, "Driving style recognition using a smartphone as a sensor platform," in *Proc. 14th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2011, pp. 1609–1615.

[204] L. M. Bergasa, D. Almería, J. Almazán, J. J. Yebes, and R. Arroyo, "DriveSafe: An app for alerting inattentive drivers and scoring driving behaviors," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2014, pp. 240–245.

[205] A. C. C. Pinilla, M. C. G. Quintero, and C. Premachandra, "Intelligent driving diagnosis based on a fuzzy logic approach in a real environment implementation," in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2014, pp. 102–107.
[206] C. Craye and F. Karray, "Driver distraction detection and recognition using RGB-D sensor," 2015, *arXiv:1502.00250*.
[207] R. A. Berri, D. R. Bruno, E. N. Borges, G. Lucca, and F. S. Osório, "ADAS classifier for driver monitoring and driving qualification using both internal and external vehicle data," in *Proc. VISIGRAPP*, 2022, pp. 560–567.

DIEGO RENAN BRUNO received the B.S. and M.S. degrees in computer science from São Paulo State University (UNESP), in 2012 and 2016, respectively, and the Ph.D. degree from the University of São Paulo (ICMC/USP), in 2020. He has been a Professor with the São Paulo College of Technology (FATEC), since 2018. His current research interests include machine learning, autonomous vehicles, robotics, intelligent driver assistance systems, computer vision, pattern recognition, and industrial robots.

RAFAEL A. BERRI received the B.S. and M.S. degrees in computer science from Santa Catarina State University (UDESC), in 2005 and 2014, respectively, and the Ph.D. degree from the University of São Paulo (ICMC/USP), in 2019. He has been a Professor with the Federal University of Rio Grande (FURG), since 2020. His current research interests include intelligent driver assistance systems, driver distraction, machine learning, financial time series, computer vision, pattern recognition, and intelligent robots.

FELIPE M. BARBOSA received the B.S. degree in computer engineering from the University of São Paulo, São Paulo, Brazil, in 2021, where he is currently pursuing the M.S. degree in computer science and computational mathematics.

He is the author of articles on RGB-D visual perception for obstacle and safe zones segmentation. His research interests include deep learning, computer vision, RGB-D and video perception, and autonomous mobile robotics.

FERNANDO S. OSÓRIO (Member, IEEE) received the B.S. and M.S. degrees in computer science from the Federal University of Rio Grande do Sul (UFRGS), in 1988 and 1991, respectively, and the Ph.D. degree from Institut National Polytechique de Grenoble, INPG, France, in 1998. He has been a Professor with the University of São Paulo (ICMC/USP), since 2008. His current research interests include machine learning, autonomous vehicles, robotics, intelligent driver assistance systems, computer vision, pattern recognition, and industrial robots.